

Asociación del Problema de Ruteo de Vehículos, con Múltiples Depósitos y Ventanas de Tiempo  
(MDVRPTW), a los Recorridos Turísticos en el Departamento de Nariño

Franklin Orlando Palacios Jácome, ✉fpalaciosj@unal.edu.co

Tesis presentada como requisito para optar al título de Magíster en Investigación Operativa y  
Estadística

Asesor: Fredy Alexander Guasmayan Guasmayan, Mg. en Investigación Operativa y Estadística



Universidad tecnológica de Pereira en convenio con la Universidad de Nariño

Facultad de ingeniería industrial

Maestría en investigación operativa y estadística

San Juan de Pasto

2018

- 
- |                      |     |  |
|----------------------|-----|--|
| Referencia/Reference | [1] | F. Palacios Jácome. “Asociación del problema de ruteo de vehículos, con múltiples depósitos y ventanas de tiempo (mdvrptw), a los recorridos turísticos en el departamento de Nariño.”, Trabajo de grado, Maestría en investigación operativa y estadística, San Juan de Pasto, Facultad de ingeniería Industrial, 2018. |
| Estilo/Style:        |     |  |
| IEEE (2014)          |     |  |
- 



En convenio con la Universidad Tecnológica de Pereira  
Maestría en investigación operativa y estadística, Cohorte XII.



### Biblioteca Universidad de Nariño



Biblioteca Digital (Repositorio)  
<http://biblioteca.udenar.edu.co/atenea/>

- Biblioteca Alberto Quijano Guerrero – Pasto
- Biblioteca Jorge Roa Martínez

### **Dedicatoria**

A Dios,  
A mis padres,  
A mi esposa,  
A mi familia,  
A mi asesor,  
Y a todo el esfuerzo dedicado a terminar este proyecto.

### **Agradecimientos**

Este proyecto es el resultado del esfuerzo conjunto,  
de las dos personas que formamos parte del grupo de trabajo.  
Por esto agradezco a mi asesor M.sc Fredy Guasmayan,  
Con quien nos hemos interesado en resaltar el departamento de Nariño,  
y darlo a conocer por medio de sus sitios turísticos,  
brindando a las personas que llegan a conocerlo,  
nuevos planes de turismo optimizados.  
A mis padres,  
quienes han apoyado mi formación personal y académica.  
A mi esposa,  
un apoyo fundamental en el proceso de este proyecto.  
A mis profesores,  
por compartir su conocimiento,  
y finalmente a estas prestigiosas Universidades,  
las cuales permitieron que varios jóvenes de nuestro departamento,  
pudiéramos acceder a esta excelente formación.

**CONTENIDO**

LISTA DE TABLAS.....	7
RESUMEN.....	10
ABSTRACT .....	12
I. INTRODUCCIÓN .....	14
II. PLANTEAMIENTO DEL PROBLEMA.....	17
2.1 Delimitación .....	19
III. JUSTIFICACIÓN.....	20
IV. OBJETIVOS .....	22
A. Objetivo general .....	22
B. Objetivos específicos .....	22
V. MARCO TEÓRICO .....	23
5.1 El problema del agente viajero (tsp): .....	23
5.2 El problema de los m – agentes viajeros (m – TSP): .....	25
5.3 El Problema con capacidades (VRP o CVRP): .....	25
5.4 El problema con ventanas de tiempo (VRPTW): .....	26
5.5 Problema de ruteo de vehículos con múltiples depósitos y ventanas de tiempo (MDVRPTW). .....	27
5.6 Heurísticas de generación de población y mejoramiento: .....	34
5.6.1 Modelo de asignación: .....	34
5.6.2 Heurística de intercambio 2-opt: .....	34
5.6.3 Heurística de ahorros Clarke y Wright: .....	37

5.6.4 Heurística del Barrido: .....	38
VI. METODOLOGÍA DE SOLUCIÓN.....	39
6.1 Codificación .....	40
6.1 Ruteo: Algoritmo Genético de Chu – Beasley modificado (AGCB) .....	47
6.2 Desarrollo metodológico .....	50
6.2.1 Asignación de clientes a los depósitos:.....	50
6.2.2 Técnicas para ruteo: .....	52
6.2.3 Intensificación algoritmo genético usando Intercambios entre vecinos cercanos. ....	58
VII. CARACTERIZACIÓN DEL CASO DE PRUEBA PARA LOS RECORRIDOS TURÍSTICOS EN EL DEPARTAMENTO DE NARIÑO. ....	58
7.1 Descripción de las rutas turísticas. ....	59
7.2 Demanda de los productos por clientes agrupados en sectores.....	62
7.3 Descripción de la matriz de costos y distancias .....	64
7.4 Vector de ventanas de tiempo .....	65
VIII. METODOLOGÍA .....	67
IX. RESULTADOS.....	70
9.1 Calibración de parámetros:.....	70
9.1.1 Asignación de clientes: .....	71
9.1.2 Evaluación y generación de la nueva población: .....	72
9.1.3 Algoritmo genético de Chu-Beasley: .....	72
9.2 Instancias .....	75
9.3 Resultados obtenidos para las instancias Cordeau .....	77

9.4 Caso de aplicación.....	80
X. CONCLUSIONES.....	89
XI. RECOMENDACIONES .....	91
XII. REFERENCIAS .....	92
XIII. ANEXOS.....	94
Anexo 1 .....	94
Anexo 2. ....	110
Anexo 3 y 4 .....	113
Anexo 5 .....	114
Anexo 6. ....	114

**LISTA DE TABLAS**

Tabla 1 .....	35
Tabla 2 .....	35
Tabla 3 .....	35
Tabla 4 .....	36
Tabla 5 .....	42
Tabla 6 .....	43
Tabla 7 .....	44
Tabla 8 .....	45
Tabla 9 .....	45
Tabla 10 .....	46
Tabla 11 .....	46
Tabla 12 .....	55
Tabla 13 .....	56
Tabla 14 .....	56
Tabla 15 .....	60
Tabla 16 .....	63
Tabla 17 .....	65
Tabla 18 .....	65
Tabla 19 .....	71
Tabla 20 .....	73
Tabla 21 .....	74
Tabla 22 .....	76
Tabla 23 .....	77
Tabla 24 .....	79
Tabla 25 .....	79
Tabla 26 .....	79
Tabla 27 .....	80
Tabla 28 .....	80
Tabla 29 .....	81

Tabla 30.....	82
Tabla 31.....	82
Tabla 32.....	83
Tabla 33.....	83
Tabla 34.....	84
Tabla 35.....	85
Tabla 36.....	86
Tabla 37.....	87
Tabla 38.....	110
Tabla 39.....	113
Tabla 40.....	114



**LISTA DE FIGURAS**

fig. 1. Esquema del VRP .....	24
fig. 2. Esquema de Solución.....	29
fig. 3. Modelo de Asignación General. ....	34
fig. 4. Distancia ahorrada .....	37
fig. 5. Diagrama de flujo para la técnica del barrido.....	38
fig. 6. Dispersión de puntos.....	43
fig. 7. Asignación de Clientes .....	47
fig. 8. Actualización de J11 .....	48
fig. 9. Actualización de J12.....	49
fig. 10. Actualización de J11 .....	50
fig. 11. Diagrama de flujo para ruteo .....	52
fig. 12. Aeropuerto Antonio Nariño Chachagüí.....	59
fig. 13. Depósito 2. Terminal Terrestre de Pasajeros Pasto. ....	59
fig. 14. Depósito 3. Terminal Transporte Terrestre Ipiales. ....	59
fig. 15. Ruta “CIRCUNVALAR AL GALERAS”.....	62
fig. 16. Diagrama de flujo del AGCB .....	72
fig. 17. Representación de la solución Pr02 de Cordeau generada por el algoritmo de solución. .	78
fig. 18. Ruta generada para el depósito 1. (San Juan de Pasto).....	82
fig. 19. Ruta generada para el depósito 2. (Chachagüí) .....	84
fig. 20. Ruta generada para el depósito 3. (Ipiales).....	86
fig. 21. Ruta 1. Generada con el código de Matlab.....	111
fig. 22. Ruta 2. Generada con el código de Matlab.....	111
fig. 23. Ruta 3. Generada con el Código de Matlab.....	112
fig. 24. Resultados Matlab.....	112

## RESUMEN

En el presente proyecto se pretende aplicar el problema de *Ruteo de vehículos con múltiples depósitos y ventanas de tiempo* (MDVRPTW), a los sitios turísticos del departamento de Nariño, este problema al igual que las otras variaciones del VRP es considerado NP-HARD, por ser muy complicado de resolver computacionalmente, por esta razón, se hace necesario para su solución la aplicación de algoritmos heurísticos y metaheurísticos.

El problema del MDVRPTW, se aborda por medio de técnicas heurísticas como herramientas de inicio para luego aplicar la metaheurística del algoritmo genético modificado propuesto por Chu-Beasley, los cuales toman soluciones dadas por métodos heurísticos como: la técnica de ahorros, técnicas de barrido, intercambios de nodos y aristas, entre otras, y luego asignarlas a una población inicial con el fin de encontrar en la composición de estas soluciones de forma aleatoria otras que mejoren la función objetivo determinada por el costo de las rutas en los vehículos [1].

El algoritmo desarrollado se prueba con diferentes instancias, en nuestro caso la Pr02, correspondiente a MDVRPTW, diseñadas por Cordeau [2], encontradas en la literatura especializada, y sus resultados son mostrados en el capítulo 6, por ser un problema relativamente pequeño por el número de clientes (33) y 3 depósitos, se prueba el algoritmo solo con una instancia Pr02 mencionada anteriormente, la cual consta de 50 clientes y 4 depósitos, esto garantiza el correcto funcionamiento del algoritmo. Después de esta comprobación se procede a aplicar el algoritmo genético simple y el AGCB para el caso particular de encontrar las mejores rutas para visitar y conocer los sitios turísticos en el departamento de Nariño.

Teniendo en cuenta que el costo de la función objetivo esta dado por la distancia que hay entre nodos, se hace necesario obtener el valor de cada sitio turístico y así generar la matriz de distancias, para ello se utiliza la aplicación de Google Maps y se construye dicha matriz de costos consultando el valor de distancia entre cada uno de los puntos considerados como turísticos, de los cuales tres de ellos serán los depósitos puesto que son los que permiten arribar al departamento en diferentes medios de transporte y desde cualquier parte del país. Adicionalmente se considera el vector de

ventanas de tiempo, el cual se forma con el tiempo mínimo requerido para visitar cada uno de ellos, además de la capacidad de cada vehículo y la demanda de cada sitio.

Con base en estos datos se genera efectuar la asignación de clientes a los depósitos, por medio del algoritmo de asignación, después con el AGS y el AGCB se genera una población para luego aplicar el ruteo a cada depósito y finalmente el mejoramiento de rutas aplicando para ello nuevamente el algoritmo genético modificado de Chu-Beasley.

**Palabras clave:** MDVRPTW, Ruteo de vehículos, Rutas Turísticas, ventanas de tiempo.

## ABSTRACT

In the present project we intend to apply the problem of vehicle routing with multiple deposits and time windows (MDVRPTW), to the tourist sites of the department of Nariño, this problem as well as the other variations of the VRP is considered NP-HARD, for being very complicated to solve computationally, for this reason, it becomes necessary for its solution the application of heuristic and metaheuristic algorithms.

The MDVRPTW problem is approached by means of heuristic techniques as starting tools to then apply the metaheuristics of the modified genetic algorithm proposed by Chu-Beasley, which take solutions given by heuristic methods such as: the savings technique, scanning techniques, exchanges of nodes and edges, among others, and then assign them to an initial population in order to find in the composition of these solutions randomly others that improve the objective function determined by the cost of routes in vehicles [1].

The developed algorithm is tested with different instances designed by Cordeau [2], found in the specialized literature, and its results are shown in chapter 6, as it is a relatively small problem due to the number of clients (33) and 3 deposits. test the algorithm with only one instance Pr02, which consists of 50 clients and 4 deposits, this guarantees the correct functioning of the algorithm. After this check we proceed to apply the metaheuristic algorithm in the particular case of finding the best routes to visit and know the tourist sites in the department of Nariño.

For this, the cost matrix is obtained by using the Google Maps application, consulting the distances between each one of the points considered as tourism, of which three of them are considered as deposits since they allow arriving at the department in different means of payment. transportation and from anywhere in the country. Additionally, the vector of time windows is considered, which is formed with the minimum time required to visit each of them, as well as the capacity of each vehicle and the demand of each site.

Based on this data, the assignment of clients to the deposits is generated, then the routing to each deposit and finally the improvement of routes applying the modified genetic algorithm of Chu-Beasley.

**Keywords:** MDVRPTW, vehicle routing, tourist routes, time windows.

## I. INTRODUCCIÓN

Descubrir Nariño es una experiencia inolvidable, conocer sus costumbres, tradiciones y sitios turísticos remite a una aventura cada vez más asombrosa y placentera. Como lo relata el poeta Aurelio Arturo en su compilado *Morada al Sur* [3]:

*“Te hablo también: entre maderas, entre resinas, entre millares de hojas inquietas, de una sola hoja: Pequeña mancha verde, de lozanía, de gracia, hoja sola en que vibran los vientos que corrieron por los bellos países donde el verde es de todos los colores, los vientos que cantaron por los países de Colombia”.*

Dicho poema hace referencia al inmenso amor del poeta por su territorio, por habitar en “el verde de todos los colores”, por haber nacido y crecido en esa tierra sin fin. Desde el mar hasta el Galeras, como dice el himno del Departamento, es un territorio en el que se encuentran fascinante fauna, flora y espectaculares pisos térmicos, lo cual invita a disfrutar de sus diferentes aromas y sabores, conjuntamente es fascinador ser espectador o partícipes de las ferias y fiestas, como máximo exponente en los primeros días de enero es nuestro Carnaval de Negros y Blancos. Es por lo anterior, y por el estar enamorado de Nariño, de sus paisajes, montañas, volcanes, ríos y mar, decidí emprender este camino e indagar por sus sitios turísticos y mejorar las condiciones de estadía de los visitantes.

Desde este punto de vista, el objetivo de este proyecto es brindar al turista, sea propio o foráneo, la optimización de las rutas que desee recorrer, de tal manera que garantice un tiempo óptimo de desplazamiento, y a su vez le permita disfrutar con mayor tranquilidad determinado lugar, minimizando el tiempo y el costo de este. Debido a la existencia únicamente de pocas rutas ya asignadas para hacer los respectivos recorridos, es necesario un trabajo como el presente ya que las rutas existentes no satisfacen la necesidad de visitar todos los puntos turísticos del departamento.

A continuación, se da a conocer el contenido del presente trabajo de investigación, el cual inicia con la descripción del problema, donde se da a conocer el planteamiento de este, la pregunta

problémica, la delimitación que a su vez demarca el espacio donde se realiza la indagación, los objetivos que por su parte son los fines que se pretenden alcanzar con este proyecto, y la justificación que sustenta la realización de esta propuesta.

Por otra parte, el diseño metodológico da a conocer los pasos que se debe seguir para encontrar una solución de calidad, efectiva y sobre todo óptima en el problema, en tanto que la revisión de antecedentes se hace para tener ideas previas que den fundamento a la investigación, de este modo el marco teórico se describen los fundamentos conceptuales que se tienen en cuenta para desarrollar la presente investigación, tales como definiciones, modelos, diagramas entre otros.

En la metodología de solución se resuelve el problema de ruteo de vehículos con múltiples depósitos y la restricción adicional de las ventanas de tiempo, empezando con la asignación de clientes a depósitos, luego con ayuda del algoritmo genético simple y las heurísticas se seleccionan los mejores individuos para ingresarlos a la población inicial y finalmente aplicar el algoritmo genético de Chu Beasley y así poder encontrar las mejores soluciones al problema.

En la caracterización del problema se describen los sitios turísticos del Departamento de Nariño, y se codifica el problema MDVRPTW con los datos de la presente investigación, es decir se explican los valores de demanda, la matriz de costos, el vector de ventanas de tiempo y los demás parámetros a tener en cuenta para encontrar la mejor solución al problema.

La metodología para solucionar el problema se lleva a cabo en diferentes fases, en una etapa inicial se estudia el problema de ruteo de vehículos, y su variación multi depósitos con ventanas de tiempo, así como las técnicas de solución, luego se formula y se implementa la metodología para finalmente validar los resultados obtenidos.

Dichos resultados se obtienen primero de la calibración y validación del algoritmo, para ello se hacen varias corridas con instancias encontradas en la literatura especializada, primero se valida la solución del VRP y el MDVRP con instancias Augerat, y luego el MDVRPTW con instancias de Cordeau.

Finalmente se dan a conocer las principales conclusiones de la investigación además de las recomendaciones que se hacen para seguir alimentando este trabajo y aplicarlo junto con algunas empresas turísticas del Departamento.



## II. PLANTEAMIENTO DEL PROBLEMA

Nariño, está ubicado en el Suroccidente del país a 841.5 kilómetros del distrito capital, por vía terrestre, es decir 17 horas aproximadamente y a 39 minutos por vía aérea. Dicho departamento se caracteriza por variedad de pisos térmicos, favoreciendo multiplicidad de climas y de cultivos, ahora bien la presente investigación está basada en la importancia turística del mismo, puesto que cuenta con un inmenso listado de sitios apetecidos por propios y visitantes, como menciona el Himno, “Desde el mar hasta el Galeras”, este es un lugar mágico, como decía el poeta Aurelio Arturo, donde “El verde es de todos los colores” y existen sin número de actividades por hacer, desde visitar el mar, realizar turismo religioso, de aventura, etno-turismo, ecoturismo, historia, cultura, entre otros [4], [5].

En este orden de ideas, haciendo un paréntesis en solo el municipio de San Juan de Pasto, que es la capital del departamento de Nariño, se puede encontrar también variedad de propuestas, entre ellas una de las más conocidas a nivel mundial y reconocidas como Patrimonio Cultural e Inmaterial de la humanidad, el Carnaval de Negros y Blancos, una fiesta cultural que alberga a miles de coterráneos y turistas, en donde se grita a viva voz “Que viva Pasto carajo”, se disfrutan de actos múltiples, dando a conocer las costumbres y tradiciones del pueblo Nariñense [6].

De acuerdo con lo anterior, al realizar la presente indagación se observó la no existencia de rutas que optimicen los recorridos y la estancia en los mismos, que satisfaga la permanencia de los turistas en el departamento y con ello garantice su pronto regreso, además de la divulgación en sus lugares de origen acerca de la relevancia de hacer turismo en Nariño.

Por lo anterior, en este proyecto, se plantea como problema la visita a diferentes sitios turísticos, y en ellos, analizar la estancia por parte de los turistas, el costo asociado a cada sitio tanto por su recorrido como el valor económico requerido para hospedaje, alimentación y transporte interno; para esto se plantea asociar el modelo de ruteo de vehículos con sus variantes de múltiples depósitos y ventanas de tiempo con el fin de usar las técnicas de la investigación de operaciones en la solución de este modelo.

Desde esta perspectiva, se tiene en cuenta términos propios tanto de turismo, investigación de operaciones, modelado y solución de modelos de optimización, entre otras herramientas así:

- **Matlab**, Es una herramienta de software matemático que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio, en el cual se realizará el algoritmo para la solución de nuestro problema.
- **Google Maps**, es una herramienta informática que muestra un globo virtual que permite volar a cualquier lugar de la Tierra para ver imágenes de satélite, mapas, imágenes de relieve y edificios 3D, o para explorar desde galaxias lejanas hasta fosas marinas. Permite explorar un detallado contenido geográfico, guardar los lugares que se han visitado y compartirlos con otros usuarios, esta herramienta será usada para mostrar los resultados obtenidos por Matlab.
- **Problema de ruteo de vehículos**, El modelo clásico del problema de ruteo vehicular conocido como VRP (vehicle route problem), describe el diseño de rutas donde a partir de un depósito del que sale cada vehículo y al que tiene que regresar, luego de visitar una sola vez a los clientes para satisfacer su demanda conocida, sin violar las restricciones de capacidad de carga de los vehículos, distancia máxima recorrida por éstos, y respetando el horario de trabajo: todo ello con el fin de buscar el costo mínimo, como se enuncia por Pirabán et al. [7].

Con base en lo anterior se desarrolla y codifica en el software Matlab el algoritmo de solución al MDVRP, y al agregar el vector de restricciones de las ventanas de tiempo, se obtiene la variante del VRP, como lo es el problema de ruteo de vehículos con múltiples depósitos y ventanas de tiempo MDVRPTW, se pretende entonces solucionar este nuevo problema con esta codificación en Matlab y con ello se genera un mapeo en Google maps, en el cual se pueda identificar las mejores rutas dependiendo del depósito, en donde se encuentren las rutas de menor distancia para el turista y con ello permitir a las personas visitar la mayor cantidad de sitios con el menor costo posible, optimizando el modelo matemático basado en la minimización del costo asociado a la distancia y cumpliendo con las restricciones propias del modelo.

## ***2.1 Delimitación***

En este trabajo se pretende resolver el problema MDVRPTW aplicado a los recorridos turísticos del departamento de Nariño, en donde existen aproximadamente 33 lugares que se pueden destinar como sitios de turismo, por ende, en muchos de ellos se encuentran más lugares subalternos por visitar por lo que se hace necesario hallar las mejores rutas haciendo uso de técnicas de solución de modelos de optimización para problemas de ruteo, tal es el caso de los métodos clásicos de programación lineal entera, como lo afirma N. Azi et al. [7]. De los sitios anteriormente mencionados, se ha determinado que 3 serán los depósitos, con los cuales se obtendrá una solución cercana a la óptima, y a la solución real.

Para representar el problema en términos de modelado matemático se realiza el siguiente proceso, primero se identifica la matriz de costos del problema, la cual se forma con la distancia medida en kilómetros, existente entre cada sitio turístico, estos valores son tomados desde la aplicación de Google Maps, considerando dicha distancia como el costo, el tiempo se considera en un vector denominado ventanas de tiempo y el dinero necesario para la estancia en cada sitio será considerado como la demanda. Posteriormente se codifica el problema para encontrar una solución que minimice los costos y así permitirle al viajero conocer la mayoría de los sitios turísticos del departamento con la menor distancia y de alguna forma garantizar también gastar la menor cantidad de dinero, puesto que, al reducir distancia recorrida según la función objetivo del modelo, se ahorra también dinero por traslado entre lugares turísticos.

### III. JUSTIFICACIÓN

Desde tiempos remotos el ser humano ha tenido que desplazarse de un lugar a otro, en búsqueda de comida, refugio o cazar, con el único fin de subsistir, convirtiendo al hombre en dichas épocas en errante; hoy en día, aunque existen muchos trotamundos, también se encuentran aquellos turistas quienes desean salir de la rutina de sus actividades cotidianas y aventurarse a explorar nuevas tierras. Ahora bien, la propuesta investigativa se enfoca en el departamento de Nariño, puesto que se pasa de la sierra al mar, en un abrir y cerrar de ojos, en tanto los aires húmedos, cálidos o fríos abrazan los amaneceres y atardeceres de propios y visitantes, deleitándose con los aromas del mejor café del mundo [8] junto a una feria gastronómica permanente, debido a la variedad de platos típicos de la región.

En los últimos tiempos, entes gubernamentales se han propuesto convertir al departamento de Nariño en un destino turístico [5] y excepcional, en el que se puedan desarrollar un sin número de actividades para todos los gustos; sin embargo, como resultado de una entrevista realizada a un encargado de la oficina de turismo departamental, se pudo analizar la inexistencia de base de datos, en las cuales se logre evidenciar información sobre:

- Clase de destino turístico favorito.
- Sitios preferidos.
- Número de turistas propios y foráneos.
- Distancias actualizadas
- Fechas y horarios de visita.
- Mapa con las mejores rutas.

En correspondencia con lo antes expuesto, al apreciar la falta de un mapa con las mejores rutas, en el que se determinarán los sitios turísticos que se podrán recorrer en el menor tiempo posible y con el mínimo costo y así poder llevar al cliente a una experiencia satisfactoria de su visita en el departamento; Y tras hacer lectura de los diferentes recursos bibliográficos donde, en fuentes nacionales se carece de investigaciones de este tipo, es notable la importancia y novedad del presente trabajo.

Ahora bien, los desplazamientos hacia dichos sitios turísticos son de fácil acceso, encontrando diversos medios de transporte para llegar a ellos, además con la firma del proceso de paz, la seguridad ha mejorado, pudiendo arribar a todo el departamento con garantías de conservación de la integridad del ser humano, durante los diversos recorridos; siendo esta una de las herramientas significativas que permiten potenciar el turismo y de igual modo la realización del presente proyecto.

Retomando lo anterior, se hace tangible la necesidad de establecer un mapa en el cual se publique a los clientes, las mejores rutas de aquellos sitios que pueden ser visitados en el menor tiempo y costo posible, de este modo se pretende ofrecer al visitante variedad de sitios turísticos y llegar a ellos minimizando los costos, por ende se asociará el problema de ruteo de vehículos con múltiples depósitos y ventanas de tiempo (MDVRPTW), a los sitios turísticos del departamento, utilizando para ello la matriz de costos del problema en la cual se consignaran las distancias entre los diferentes sitios de turismo y se codificaran para hallar una buena solución cercana a la óptima.

El MDVRPTW es una variación del problema de ruteo de vehículos VRP, en donde se consideran los múltiples depósitos, es decir, varios sitios desde donde se ubican los vehículos que van a visitar a los clientes, y a los cuales deben retornar una vez terminado el recorrido [9], en este caso serán las ciudades a las cuales hay acceso desde cualquier otra parte del país, ciudades principales con acceso por vía aérea y terrestre, con la restricción adicional de que asocia a cada cliente una ventana de tiempo, dichas ventanas son intervalos de tiempo en el cual cada cliente es atendido, si el vehículo llega antes del tiempo especificado en la ventana, éste deberá esperar a ser atendido, en esta variación del VRP, el objetivo es minimizar la distancia recorrida entre las rutas, teniendo en cuenta el tiempo total de espera al suministro de los clientes [10].

## IV. OBJETIVOS

### *A. Objetivo general*

Solucionar el MDVRPTW mediante la implementación de un algoritmo genético modificado de Chu-Beasley con asociación al turismo en el departamento de Nariño

### *B. Objetivos específicos*

Describir los parámetros y restricciones pertinentes al modelo en el contexto del turismo del departamento de Nariño

Determinar características propias del modelo matemático y variables que lo alimentan

Establecer las rutas turísticas óptimas y presentarlas en un plan turístico para los clientes.

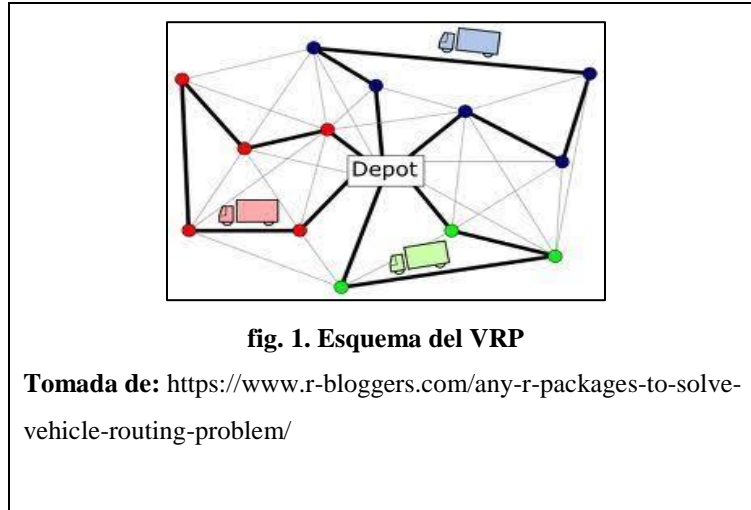
## V. MARCO TEÓRICO

El Problema de Ruteo de Vehículos es una herramienta fundamental en el desarrollo de las empresas, puesto que permite dar un aporte importante en la reducción de costos al momento de realizar las entregas de los productos a los diferentes clientes, por medio de las rutas de costo mínimo que comienzan y terminan en los depósitos.

El ruteo de vehículos corresponde a un tipo de problemas combinatoriales que en la medida en que se pretenda obtener la o las rutas optimas que permitan atender completamente un conjunto de clientes determinado, el problema se vuelve muy complicado computacionalmente, de forma que para un número de clientes mayor a 20 con una sola ruta se tiene  $20!/2$  combinaciones posibles, es decir  $12 \times 10^{17}$  combinaciones, siendo esto casi imposible de resolver computacionalmente con los métodos exactos existentes, los cuales resuelven el problema por medio del modelado matemático, es decir buscando cada una de las combinaciones posibles y teniendo en cuenta cada una de las restricciones del problema, como se indica en [1].

### ***5.1 El problema del agente viajero (tsp):***

En el problema del Agente Viajero (o TSP por Travelling Salesman Problem) se dispone de un solo vehículo el cual debe recorrer varias ciudades y regresar al depósito desde el cual hace su partida en la ruta de costo mínima, se considera como uno de los problemas de ruteo más simples, puesto que de él se desprenden muchas generalizaciones del TSP.



El problema puede formularse [11] como:

$$\min \sum_{(i,j) \in E} c_{ij} X_{ij} \quad 3.1$$

$$s. a. \sum_{j \in \Delta^+(i)} X_{ij} = 1 \quad \forall i \in V \quad 3.2$$

$$\sum_{j \in \Delta^-(i)} X_{ij} = 1 \quad \forall j \in V \quad 3.3$$

$$\sum_{i \in S, j \in \Delta^+(i)} X_{ij} \geq 1 \quad \forall S \subset V \quad 3.4$$

$$X_{ij} \in \{0, 1\} \quad \forall (i, j) \in E \quad 3.5$$

Las variables binarias  $X_{ij}$  indican si el arco  $(i, j)$  es utilizado en la solución. La función objetivo 3.1, establece que el costo total de la solución es la suma de los costos de los arcos utilizados. Las restricciones 3.2 y 3.3 indican que la ruta debe llegar y abandonar cada nodo exactamente una vez. La restricción 3.4 se denomina de eliminación de sub-tours e indican que cada subconjunto de nodos  $S$  debe ser abandonado al menos una vez



### 5.2 El problema de los $m$ – agentes viajeros ( $m$ – TSP):

Este problema es una generalización del TSP, en él existe un número finito  $m$  de agentes los cuales deben cubrir diferentes ciudades y que cada ciudad sea visitada por un único agente, el objetivo es construir exactamente  $m$  rutas para cada agente.

### 5.3 El Problema con capacidades (VRP o CVRP):

El problema de ruteo de vehículos VRP, es una extensión del ( $m$  – TSP) donde existe una demanda asociada a cada ciudad y una capacidad determinada para cada uno de los vehículos, cuyo objetivo es la disminución del costo de recorrido de los vehículos en todos los trayectos. [1]. El modelo matemático para el VRP según [12] es:

$$\min \sum_{(i,j) \in E} C_{ij} X_{ij} \quad 3.6$$

$$s.a. \sum_{j \in \Delta^+(0)} X_{0j} = m \quad 3.7$$

$$\sum_{j \in \Delta^-(0)} X_{i0} = m \quad 3.8$$

$$\sum_{j \in \Delta^+(i)} X_{ij} = 1, \forall i \in V \setminus \{0\} \quad 3.9$$

$$\sum_{j \in \Delta^-(i)} X_{ij} = 1, \forall j \in V \setminus \{0\} \quad 3.10$$

$$\sum_{i \in S, j \in \Delta^+(i) \setminus S} X_{ij} \geq r(S), \forall S \subset V \setminus \{0\} \quad 3.11$$

$$m \geq 1$$

$$X_{ij} \in \{0,1\}, \forall (i,j) \in E$$

La función objetivo 3.6 es el costo total de la solución. Las restricciones 3.7 y 3.8 indican que  $m$  es la cantidad de vehículos utilizados en la solución y que todos los vehículos que parten del depósito deben regresar. Las restricciones 3.9 y 3.10 aseguran que todo cliente es un nodo intermedio de alguna ruta. Finalmente, la restricción 3.11 actúa como restricción de eliminación de

sub-tours y a la vez impone que la demanda total de los clientes visitados por un vehículo no puede superar la capacidad  $C$ .

Determinar el valor de  $r(S)$ , requiere de la solución del siguiente problema:

$$r(S) = \min \sum_{k \in K} y_k \quad 3.12$$

$$\text{s.a. } \sum_{i \in S} d_i x_{ik} \leq C_{y_k}, \forall k \in K \quad 3.13$$

$$\sum_{k \in K} x_{ik} = 1, \forall i \in S \quad 3.14$$

$$x_{ik} \in \{0,1\}, \forall i \in S, \forall k \in K$$

$$y_k \in \{0,1\}, \forall k \in K$$

Donde,  $K$  es un conjunto de vehículos que satisfacen la demanda  $n$ . En esta formulación, se asume que la disponibilidad de vehículos es ilimitada. Si se tuviera una flota finita se podría agregar una cota superior para  $m$  o directamente fijar su valor. Está implícito que cada vehículo puede recorrer a lo sumo a una ruta.

#### 5.4 El problema con ventanas de tiempo (VRPTW):

Este es otra variación del problema de ruteo de vehículos en el cuál además de capacidades, a cada cliente se le asocia una ventana de tiempo, en ella se establece un horario de permanencia y de arribo al cliente. El problema se formula de la siguiente manera [13]:

$$\min \sum_{k \in K} \sum_{(i,j) \in E} c_{ij}^k x_{ij}^k \quad 3.15$$

$$\text{s.a. } \sum_{k \in K} \sum_{j \in \Delta^-(i)} x_{ij}^k = 1, \forall i \in V \setminus \{0, n+1\} \quad 3.16$$

$$\sum_{j \in \Delta^+(0)} x_{0j}^k = 1, \forall k \in K \quad 3.17$$

$$\sum_{j \in \Delta^+(i)} x_{ij}^k - \sum_{j \in \Delta^-(i)} x_{ji}^k = 0, \forall k \in K, i \in V \setminus \{0, n+1\} \quad 3.18$$

$$\sum_{i \in V \setminus \{0, n+1\}} d_i \sum_{j \in \Delta^+(i)} x_{ij}^k \leq q^k, \forall k \in K \quad 3.19$$

$$y_j^k - y_i^k \geq s_i + t_{ij}^k - M(1 - x_{ij}^k), \forall i, j \in V \setminus \{0, n+1\}, k \in K$$

$$e_i \leq y_i^k \leq l_i, \forall i \in V \setminus \{0, n+1\}, k \in K$$

$$x_{ij}^k \in \{0, 1\}, \forall (i, j) \in E, k \in K$$

$$y_i^k \geq 0, \forall i \in V \setminus \{0, n+1\}, k \in K$$

Las variables  $x_{ij}^k$ , indican si el arco  $(i, j)$  es recorrido por el vehículo  $k$ . Las variables  $y_i^k$ , indican la hora de arribo al cliente  $i$  cuando es visitado por el por el vehículo  $k$ . (si el cliente no es visitado por dicho vehículo el valor de la variable no tiene significado). La función objetivo 1.12, es el costo total de las rutas. La restricción 3.15, indica que todos los clientes deben ser visitados. Las restricciones 3.16 y 3.17 determinan que cada vehículo  $k \in K$  recorre un camino de 0 a  $n + 1$ . En 3.18 se determina la capacidad de cada vehículo. La restricción 3.19 asegura que si un vehículo  $k$  viaja de  $i$  a  $j$ , no puede llegar antes que  $y_i + s_i + t_{ijk}$ , y actúan además como restricciones de eliminación de sub-tours. ( $M$  suficientemente grande). En 3.19 se imponen los límites de las ventanas de tiempo.

### **5.5 Problema de ruteo de vehículos con múltiples depósitos y ventanas de tiempo (MDVRPTW).**

El MDVRPTW consiste en asignar rutas a vehículos que se encuentran dispersos en un área y en el cual cada vehículo se encuentra ubicado en uno de muchos depósitos desde el cuál este debe iniciar su recorrido y al cual debe retornar una vez terminada la ruta asignada al periodo [9].

Por otra parte, las ventanas de tiempo se producen cuando los clientes requieren que el servicio de entrega o de recogida se produzca dentro de una franja horaria especificada, la cual viene determinada por una hora temprana y otra tardía de servicio. Igualmente, se puede incluir un límite

en el tiempo total de conducción de los vehículos debido a las regulaciones existentes en los contratos de los conductores [14][9].

En este orden de ideas, la solución del problema de ruteo de vehículos y cualquiera de sus variantes se puede realizar mediante la aplicación de métodos exactos y heurísticas; en tanto los métodos exactos se aplican para resolver problemas en los cuales el número de clientes es de menos de 70 aproximadamente esto ya que si hay más se torna un problema computacionalmente muy complejo de resolver.

De este modo para problemas en los cuales el número de clientes es mayor se hace necesario el uso de métodos de solución conocidos como las heurísticas y las metaheurísticas las cuales son poderosas herramientas computacionales utilizadas en la solución de problemas de planeación de rutas turísticas.

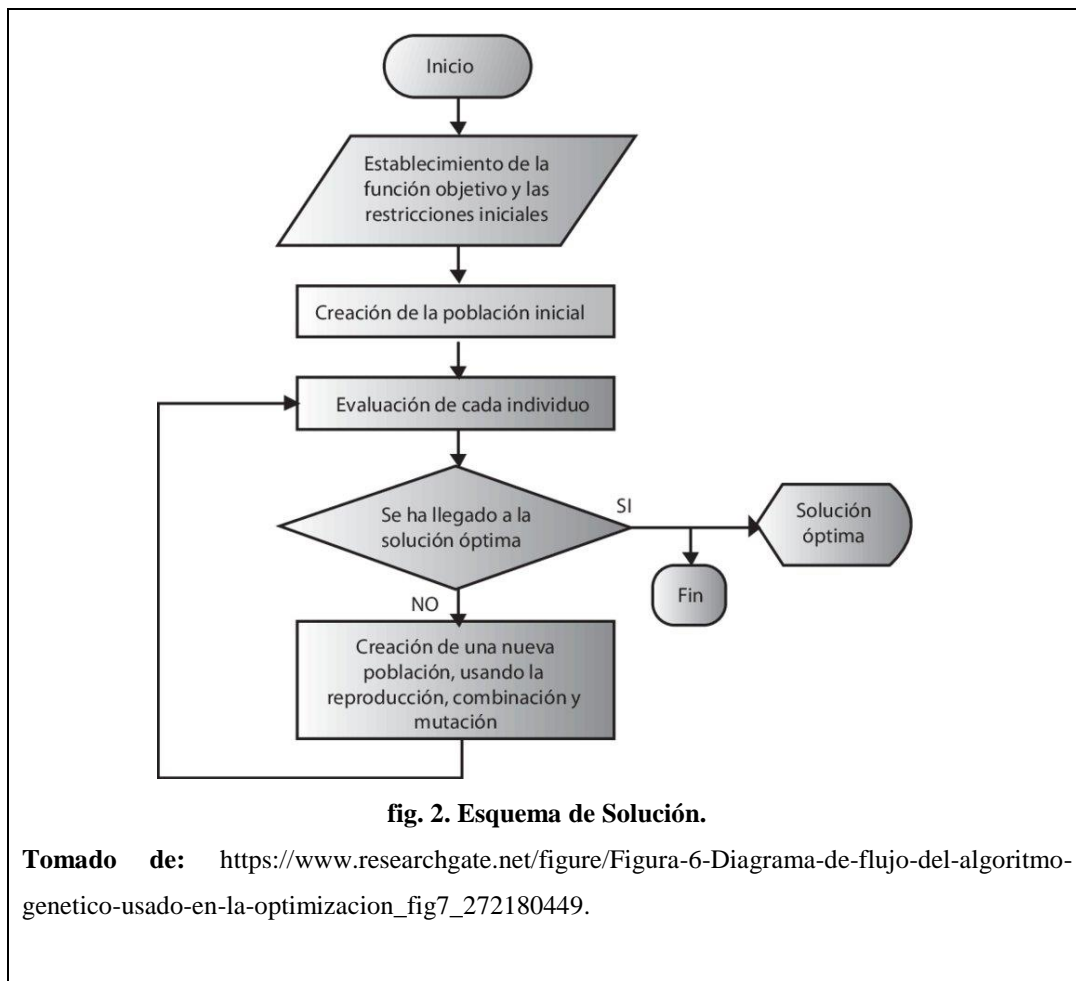
El modelo puede formularse como [14][13]:

Las siguientes suposiciones se utilizan en el problema de ruteo de vehículos con múltiples depósitos y ventana de tiempo:

- (1) Cada vehículo debe salir y regresar al mismo depósito
- (2) Cada cliente es visitado una vez y por un vehículo
- (3) La demanda del cliente a lo largo de la ruta no excede la capacidad del vehículo.
- (4) Todos los clientes con demandas conocidas están asignados a vehículos.
- (5) El tiempo total de una ruta no excede el tiempo máximo de ruta del vehículo. Los tiempos máximos de ruta del vehículo son equivalentes a la ventana de tiempo del depósito
- (6) Las restricciones de la ventana de tiempo de cada entrega del cliente deben ser satisfechas.

- (7) La diferencia entre los tiempos de llegada de los vehículos de entrega e instalación en el mismo cliente no debe exceder el nivel de servicio (intervalo de tiempo máximo permitido).

A continuación, se presenta el diagrama de flujo que representa el modelo de solución que se va a utilizar en la presente investigación.



La siguiente notación es usada en el modelo de MDVRPTW:

Conjuntos:

$I$  : conjunto de nodos de entrega.

$A$  : conjunto de nodos de instalación.  $A \subset I$

$J$  : conjunto de nodos depósitos.

$N$  : conjunto de todos los nodos.  $N = I \cup J$

$L$  : conjunto de nodos de instalación y depósitos.  $L = A \cup J$

$K$  : conjunto de vehículos de entrega.

$S$  : conjunto de vehículos de instalación.

Parámetros:

$F_j$  : Costo fijo para el depósito  $j$ .

$CF_k$  : Costo fijo para el vehículo  $k$ .

$CF_s$  : Costo fijo para el vehículo  $s$ .

$CT_k$  : Costo de transporte para el vehículo  $k$  por unidad de tiempo.

$CT_s$  : Costo de transporte para el vehículo  $s$  por unidad de tiempo.

$CR_k$  : Costo de mano de obra de la persona a cargo del vehículo  $k$  por unidad de tiempo.

$CR_s$  : Costo de mano de obra de la persona a cargo del vehículo  $s$  por unidad de tiempo.

$t_{ij}$  : Tiempo de transporte entre los nodos  $i$  y  $j$ .

$st_i$  : Tiempo de servicio del vehículo de instalación del cliente  $i$ .

$e_i$  : Tiempo más temprano en el cliente  $i$ .

$l_i$  : Ultimo tiempo en el cliente  $i$ .

$r_k$  : Tiempo máximo de ruta permitido para el vehículo  $k$ .

$d_i$  : Demanda en el cliente  $i$ .

$q_k$  : Capacidad del vehículo  $k$ .

$SL$  : Nivel de servicios para la instalación de servicios

$M$  : Un numero suficientemente grande

Variables de decisión:

$a_i$ : Hora de llegada del vehículo al cliente  $i$

$b_i$ : Hora de llegada del vehículo de instalación al cliente  $i$

$a_{jk}$ : Hora de llegada del vehículo de entrega  $k$  en el deposito  $j$

$b_{js}$ : Hora de llegada del vehículo de instalación  $s$  en el deposito  $j$

$wd_i$ : Tiempo de espera del vehículo de entrega al cliente  $i$

$wt_i$ : Tiempo de espera del vehículo de instalación a cliente  $j$

$$x_{ijk} = \begin{cases} 1, & \text{Si el vehículo de entrega } k \text{ viaja directamente de } i \text{ a } j \\ 0, & \text{otro caso} \end{cases}$$

$$\forall i, j \in N, k \in K.$$

$$x_{ijs} = \begin{cases} 1, & \text{Si el vehículo de instalación } s \text{ viaja directamente de } i \text{ a } j \\ 0, & \text{otro caso} \end{cases}$$

$$\forall i, j \in L, s \in S.$$

$$Z_j = \begin{cases} 1, & \text{Si el depósito } j \text{ es usado.} \\ 0, & \text{otro caso} \end{cases}$$

$$\forall j \in J$$

La formulación del MDVRPTW, puede formularse de la siguiente manera.

$$\min \sum_{j \in J} F_j Z_j + \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} CT_k t_{ij} x_{ijk} + \sum_{i \in L} \sum_{j \in L} \sum_{s \in S} CT_s t_{ij} x_{ijs} + \quad 3.20$$

$$\sum_{k \in K} \sum_{i \in J} \sum_{j \in I} CF_k x_{ijk} + \sum_{s \in S} \sum_{i \in J} \sum_{j \in A} CF_s x_{ijs} + \sum_{j \in J} \sum_{k \in K} CR_k a_{jk} + \sum_{i \in J} \sum_{s \in S} CR_s b_{js}$$

$$s.a \sum_{i \in N} x_{ijk} = \sum_{i \in N} x_{jik}, \forall j \in N, k \in K. \quad 3.21$$

$$\sum_{i \in L} x_{ijs} = \sum_{i \in L} x_{jis}, \forall j \in L, s \in S, \quad 3.22$$

$$\sum_{k \in K} \sum_{i \in N} x_{ijk} = 1, \forall j \in L, \quad 3.23$$

$$\sum_{s \in S} \sum_{i \in L} x_{ijs} = 1, \forall j \in A, \quad 3.24$$

$$\sum_{i \in J} \sum_{j \in I} x_{ijk} \leq 1, \forall k \in K, \quad 3.25$$

$$\sum_{i \in J} \sum_{j \in A} x_{ijs} \leq 1, \forall s \in S, \quad 3.26$$

$$\sum_{i \in N} x_{ijk} \leq Z_j, \forall j \in J, k \in K, \quad 3.27$$

$$\sum_{i \in L} x_{ijs} \leq Z_j, \forall j \in J, s \in S, \quad 3.28$$

$$\sum_{i \in I} \sum_{j \in N} d_i x_{ijk} \leq q_k, \forall k \in K, \quad 3.29$$

$$a_j \geq a_i + wd_i + t_{ij} + M \left( \sum_{k \in K} x_{ijk} - 1 \right), \forall i \in N, j \in I, \quad 3.30$$

$$b_j \geq b_i + st_i + wt_i + t_{ij} + M \left( \sum_{s \in S} x_{ijs} - 1 \right), \forall i \in L, j \in A, \quad 3.31$$

$$r_k \geq a_{jk} \geq a_i + wd_i + t_{ij} + M (x_{ijk} - 1), \forall i \in I, j \in J, k \in K \quad 3.32$$

$$r_s \geq b_{js} \geq b_i + st_i + wt_i + t_{ij} + M (x_{ijs} - 1), \forall i \in A, j \in J, s \in S \quad 3.33$$

$$e_i \leq a_i + wd_i \leq l_i, \forall i \in I, \quad 3.34$$

$$a_i + wd_i \leq b_i + wt_i \leq SL + a_i + wd_i, \forall i \in A, \quad 3.35$$

$$a_i = b_j = st_j + wd_i + wt_i = 0, \forall j \in J, \quad 3.36$$

$$x_{ij} = \{0, 1\}, \forall i, j \in N, k \in K, \quad 3.37$$

$$x_{ijs} = \{0, 1\}, \forall i, j \in L, s \in S, \quad 3.38$$

$$Z_j = \{0, 1\}, \forall j \in J. \quad 3.39$$

La función objetivo (3.20), representa el costo total, el cual se puede interpretar como el tiempo de viaje o distancia recorrida total, incluidos los costos fijos de los depósitos y los vehículos de entrega e instalación, así como los relacionados con el transporte y la mano de obra. Los costos fijos se amortizan para que puedan compararse directamente con los costos de operación.

Las Restricciones (3.21) y (3.22) aseguran que el vehículo que deja al cliente, es el mismo vehículo que el que visita al cliente.



Las Restricciones (3.23) y (3.24) estipulan que cada cliente está asignado a un vehículo de entrega y que cada cliente que necesita instalación está asignado a exactamente un vehículo de instalación, respectivamente.

Las restricciones (3.25) y (3.26) definen si cada vehículo debe asignarse o no a un depósito.

Las Restricciones (3.27) y (3.28) garantizan que los vehículos se pueden asignar a un depósito solo si ese depósito está abierto.

Restricción (3.29) es la restricción de capacidad de los vehículos de reparto. Las restricciones (3.30) – (3.33) garantizan tiempos de llegada compatibles para el servicio, la espera y el transporte entre los nodos  $i$  y  $j$ . El tiempo de servicio se considera solo para un vehículo de instalación. Las Restricciones (3.34) y (3.35) especifican las ventanas de tiempo en cada cliente, y el nivel de servicio es el intervalo de tiempo entre los servicios de entrega e instalación.

Restricción (3.36) define los tiempos de depósito. Finalmente, las Restricciones (3.37), (3.38) y (3.39) son los requisitos binarios para las variables de decisión con respecto a la ruta del vehículo de entrega, la ruta del vehículo de instalación y la ubicación del depósito, respectivamente.

Para nuestro caso de investigación se solucionará el MDVRP, propuesto en [17], en donde la función objetivo consiste en minimizar la distancia recorrida total de cada una de las rutas, anexando la restricción de un nuevo vector en el cual se consideran los tiempos de permanecía en cada uno de los nodos, llamado vector de ventanas de tiempo.

Es decir, la función objetivo es:  $\min \sum_{(i,j) \in E} C_{ij} X_{ij}$  ; la cual representa el costo total de las rutas

s.a  $\sum_{(i,j) \in w} t_{ij} \leq T$  ; donde  $t_{ij}$  son los tiempos de permanencia en cada nodo y

$T$  es el máximo recurso que se cuenta en cada ruta

### 5.6 Heurísticas de generación de población y mejoramiento:

Las heurísticas que se usan para generar individuos a la población inicial del algoritmo genético son el algoritmo de asignación, el algoritmo 2-opt [1].

#### 5.6.1 Modelo de asignación:

El algoritmo modela un problema de asignación como una matriz de costos  $n \times m$ , donde cada elemento representa el coste de asignar el  $n$ -ésimo trabajador al  $n$ -ésimo trabajo. Por defecto, el algoritmo realiza la minimización de los elementos de la matriz; de ahí que en caso de ser un problema de minimización de costes, es suficiente con comenzar la eliminación de Gauss-Jordan para hacer ceros (al menos un cero por línea y por columna), el cual es el método conocido como método Húngaro.

Sin embargo, en caso de un problema de maximización del beneficio, el costo de la matriz necesita ser modificado para que la minimización de sus elementos lleve a una maximización de los valores de costo originales.

El modelo de asignación es un problema de transporte donde cada una de las personas es el origen y cada una de las tareas son los destinos. La oferta y demanda son igual a uno, figura 3, es decir, cada origen tiene una sola persona y cada destino necesita sólo una persona [15].

ORIGEN	DESTINO				OFERTA
		1	2	3	
A		C11	C12	C13	1
B		C21	C22	C23	1
C		C31	C32	C33	1
	DEMANDA	1	1	1	

**fig. 3. Modelo de Asignación General.**

**Tomado de:** <https://www.gestiopolis.com/modelo-asignacion-caso-modelo-transporte/>

La figura 3, muestra el modelo de asignación general.

**Ejemplo:** Se tienen 3 máquinas para realizar 3 tareas, se pide definir la asignación óptima de minimización.

La tabla 1, muestra la matriz de los datos en pesos requeridos para hacer las tareas.

**Tabla 1**

**Matriz de Datos**

Maquinas/tareas	1	2	3
A	49	86	54
B	45	79	66
C	46	58	78

Fuente: <https://www.gestiopolis.com/modelo-asignacion-caso-modelo-transporte/>

En este caso tenemos una tabla balanceada, puesto que el número de columnas y filas es igual, entonces, elegimos el menor valor de cada fila, en nuestro caso son 49, 45 y 46. Restamos estos valores a su fila correspondiente, y obtenemos los valores de la tabla 2.

**Tabla 2**

**Resta del menor valor a cada fila**

Maquinas/tareas	1	2	3
A	-5	32	0
B	0	34	21
C	0	12	32

Ahora hacemos el mismo proceso, pero por columna, es decir escogemos el menor valor de cada columna y lo restamos a los valores de ellas, se obtienen los datos de la tabla 3.

**Tabla 3**

**Resta por Columna y cancelación de ceros**

Maquinas/tareas	1	2	3
A	-5	20	0
B	0	22	21
C	0	0	32

La tabla 3, muestra también la cancelación de ceros por medio de las líneas de colores, y observamos que son 3, por lo cual el modelo es óptimo. Ahora se realizan las asignaciones, esto se muestra en la tabla 4, en donde los de color café son las asignaciones y el valor en color verde no se toma para la asignación.

**Tabla 4**  
**Asignación**

Tareas/Maquinas	1	2	3
A	-5	20	0
B	0	22	21
C	0	0	32

Se traduce la solución, la tarea A se realiza en la máquina 3 con un costo de \$54, la B en la 1 con un costo de \$45 y la C en la 2 con un costo de \$58, de donde se tiene que el costo total mínimo es \$157

#### **5.6.2 Heurística de intercambio 2-opt:**

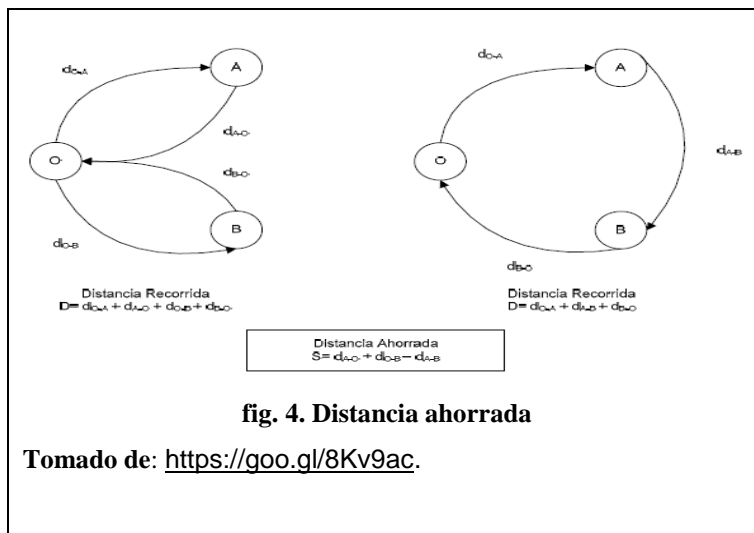
El algoritmo consiste básicamente en eliminar dos aristas y conectar los dos caminos resultantes, de una manera diferente, para obtener un nuevo ciclo [16]. Para el caso particular del problema se toma un vector desde 2 hasta  $n$ , donde  $n$  representa el número de clientes que se debe visitar y el numeral 1 representa el depósito, por lo que no se indica en el vector, este vector se permuta para formar una ruta aleatoria y se genera para este las rutas respectivas de acuerdo con la capacidad de los vehículos y la demanda de los distintos clientes. De igual manera se obtiene el valor objetivo de cada ruta con el fin de verificar si los intercambios mejoran la calidad de las respuestas en función de la matriz de distancias [17].

**Ejemplo:** Esencialmente el algoritmo se puede evidenciar de la siguiente manera, si tenemos una ruta: A - B - C - D - E - A, se reemplaza (A, B) y (C, D) por (A, C) y (B, D) y se obtiene una nueva ruta con "2-cambios" : A - C - B - D - E - A

### 5.6.3 Heurística de ahorros Clarke y Wright:

El algoritmo de ahorros tiene como objetivo minimizar el número de vehículos a utilizar para visitar a todos los clientes, así como el de minimizar la distancia total recorrida por cada uno de ellos, mediante esta heurística de ahorro se pretende unir las rutas combinando dos destinos y con ello generar una nueva ruta en la cual se evidencie el ahorro en el costo de esta. Finalmente se evalúa la ruta antes y después de la combinación para determinar el ahorro o la distancia ahorrada [18].

En la figura 4, se muestra el esquema y la fórmula que permite hallar la distancia ahorrada aplicando la combinación de dos clientes.



Donde  $S$  representa la distancia ahorrada al introducir la ruta  $A-B$ , la variable  $d$ , representa el costo de ir desde  $A$  hasta  $B$ . Para que la nueva ruta sea válida, no debe violar las restricciones del problema, es decir la capacidad del vehículo, el sentido de las rutas y otras más.

Una de las deficiencias del algoritmo de Clark & Wright es que tiende a producir buenas rutas al comienzo, pero rutas menos interesantes hacia el final. Para remediar esto, se puede realizar una penalización en la función de ahorros con la unión de rutas con clientes lejanos [1].

#### 5.6.4 Heurística del Barrido:

Esta técnica consiste en hacer un barrido angular con la ayuda de una semirrecta con origen en cada depósito hacia los clientes, estos se asignan hasta que se viole la restricción de capacidad de cada vehículo. Para después rutear los nuevos *clusters* resolviendo el Tsp.

Esta técnica se puede aplicar en tres pasos, en los cuales en el primero se hace la ordenación de los clientes según el ángulo de manera creciente. Si se presenta el caso de dos clientes que tengan el mismo valor para el ángulo, se escoge el de menor radio. Luego de esto, se van agregando los clientes que estén en el barrido de la semirrecta sin violar la capacidad. Y finalmente se resuelve un Tsp para los clientes seleccionados[19].

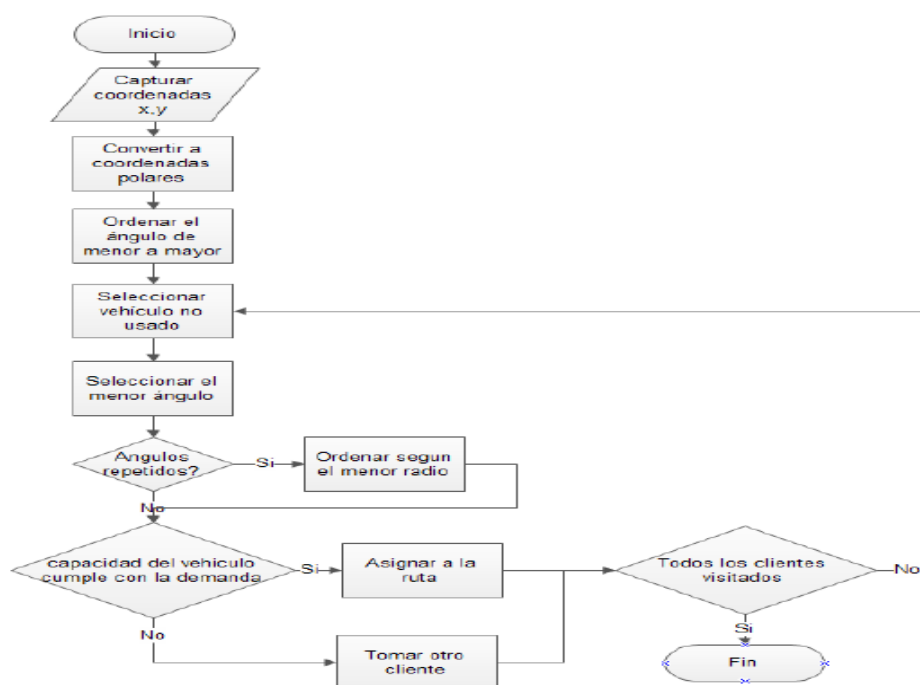


fig. 5. Diagrama de flujo para la técnica del barrido

Tomado de: <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/5663/621319L735.pdf?sequence=3>

## VI. METODOLOGÍA DE SOLUCIÓN.

Para solucionar los problemas de ruteo de vehículos, debido a su complejidad NP-hard, se hace necesario implementar métodos de solución apropiados, los cuales se aplican según el tamaño de la dificultad (número de nodos), de esta manera en inconvenientes pequeños se pueden aplicar métodos exactos los cuales permiten dar solución al mismo, pero si el problema aumenta de tamaño estos métodos son inapropiados ya que a mayor número de individuos mayor es la complicación en tiempo computacional, para esto se implementan los métodos heurísticos y meta heurísticos.

En esta sección se presentará una metodología híbrida que permita encontrar soluciones factibles por medio de algoritmos genéticos, en los cuales se tomarán algunos individuos aleatoriamente y otros se encontrarán aplicando métodos de solución heurísticos que según Bransord y Stein (1984) este método incluye cinco pasos: identificar el problema, definir y presentar el problema, explorar las estrategias viables, avanzar en las estrategias y lograr la solución y volver para evaluar los efectos de las actividades, y métodos metaheurísticos, para con ellos generar la población inicial, esto con el fin de cumplir con uno de los puntos importantes del algoritmo genético. AG.

En seguida, se presentan los pasos que conforman la solución:

En primera instancia se toma las características del MDVRPTW (coordenadas, demanda, capacidad, número de depósitos, ventanas de tiempo). Los principales datos para desarrollar el problema. Aquí se establecerán los clientes a depósitos, mediante el modelo de la asignación, el cual está validado con instancias CORDEAU [2], como se indica en [17], por ello se asignará clientes cercanos hasta terminar los recursos.

A continuación, se realiza la partición de vectores de acuerdo con la demanda, pertenencia a depósito, secuencia de vehículos y rutas entre otros, la estructura con estas características se denominará individuo y éstos son enviados al Algoritmo Genético de Chu Beasley (AGCB), para generar el ruteo por depósito, considerando varias soluciones de asignación en cada depósito se generan una o más rutas según la demanda del cliente y la capacidad del vehículo. Posteriormente, se mejoran las soluciones obtenidas a través de algoritmos de intercambio, tanto entre rutas como

entre depósitos, utilizando los algoritmos de intercambios, 2-opt y Shift (1,0) y (2,0). Por último, la solución será la mejor estructura vectorial, junto con su representación gráfica.

### 6.1 Codificación

Los siguientes elementos son vectores que permiten codificar el problema de esta investigación y se presentan en el ejemplo 2, el funcionamiento del algoritmo.

Secuencia de visitas de clientes (**J**):

El cliente (**C**) es cada uno de los 33 sitios turísticos del departamento de Nariño.

$c_j$ : cliente  $j$  –ésimo, donde  $j \in \{1,2,3, \dots, 33\}$

**J**: Vector de Clientes,  $J = [c_1, c_2, \dots, c_n]$ , donde  $n$  es el número de clientes.

Demanda (**D**): es el dinero mínimo que debo gastar en cada uno de los sitios turísticos:  $d_j$  para  $j = 1,2,3, \dots, n$ .

$D = [d_1, d_2, d_3, \dots, d_n]$ , Siendo  $d_1$ , la demanda para el cliente 1.

Capacidad es el dinero que tiene cada turista para visitar los puntos.  $cap = [q_1, q_2, q_3, \dots, q_n]$

Número de clientes asignados a cada depósito (**Depo**):  $Depo = [a_1, a_2, a_3, \dots, a_d]$ , donde  $d$  es el número de depósitos.

Entonces  $a_1$  es el número del primer cliente asignado al depósito 1.

$a_i$ : Representa a cada elemento del vector **Depo**, el cual contiene el número de clientes asignado al depósito  $i$ .

Rutas o vehículos (**V**): Matriz de dimensión igual al número de depósitos por el número máximo de rutas por depósito + 1.

$$V = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1k} & 0 \\ b_{21} & b_{22} & \cdots & b_{2k} & 0 \\ b_{d1} & b_{d2} & \cdots & b_{dk} & 0 \end{pmatrix}$$

Donde  $k$ , es el número de vehículos por ruta, y  $b_{dk}$  es el número de clientes desde el depósito  $d$  hasta  $k$  vehículos, por ejemplo  $b_{11}$ , es el número de clientes de la primera ruta del depósito 1.



Función objetivo ( $f_o$ ): La función objetivo es minimizar el costo, el cual consta de la suma de costos de ir de un sitio a otro.

$$f_o = [Q]$$

Donde, cada  $Q_i$  = Costo total de las rutas de todos los depósitos, los valores  $Q_i$  se obtienen con la suma de valores de la función objetivo de cada ruta, luego, sumando cada función objetivo de cada ruta se obtiene la función objetivo de cada uno de los depósitos y finalmente la suma de los depósitos es la función objetivo total.

La siguiente matriz representada por la variable A es una variable que almacena un conjunto de datos de diferente tipo por eso se considera como variable tipo estructura ya que esta tiene asociadas las variables de clientes, demandas, depósitos, rutas y funciones objetivos de cada ruta.

$$A = \begin{pmatrix} A.J_1 & A.D_1 & A.Depo_1 & A.V_1 & A.f_{o1} \\ A.J_2 & A.D_2 & A.Depo_2 & A.V_2 & A.f_{o2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A.J_m & A.D_m & A.Depo_m & A.V_m & A.f_{om} \end{pmatrix}$$

Donde  $m$ , es el número de soluciones que se generan en el algoritmo genético

Se debe tener en cuenta que en la matriz A. J es un vector, D es una matriz de demandas, V es una matriz de rutas de cada depósito, y f es una columna, por tanto, la matriz A no es cuadrada, opuesto que es una matriz que contiene varias matrices de diferentes dimensiones.

Y finalmente se considera el vector de las ventanas de tiempo  $tw$ , el cual es un vector formado por los tiempos mínimos requeridos en cada sitio turístico.

$$tw = [t_1, t_2, t_3, \dots, t_n]$$

En el problema, cada turista se lo considera como un vehículo, puesto que son quienes realizaran los recorridos por cada una de las rutas.

### Ejemplo:

A continuación, se presenta un ejemplo simplificado, en el cual se da a conocer el proceso del algoritmo y la codificación:

El ejemplo se presenta con las siguientes características:

Número de vehículos,  $m = 3$ , para nuestro caso son los turistas,

Número de clientes,  $n = 8$ .

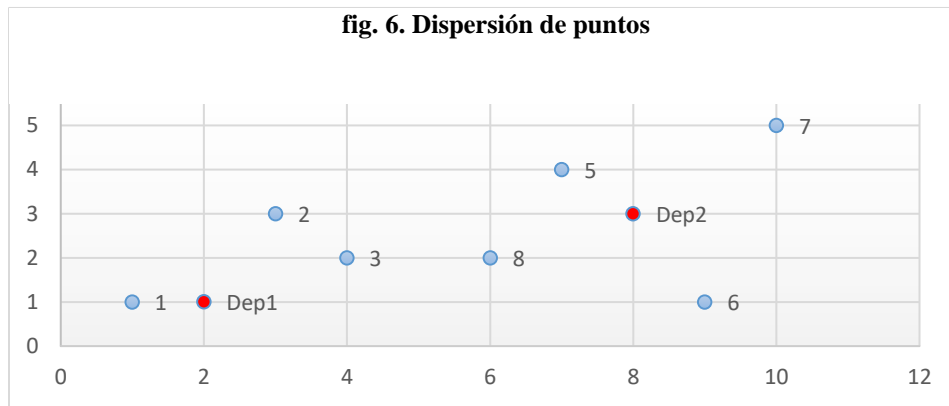
Dos depósitos,  $d = 2$ .

Los datos de entrada se presentan en la tabla 5.

**Tabla 5**  
**Datos de entrada**

<b>Sitio turístico</b>	<b>Coordenada x</b>	<b>Coordenada y</b>	<b>Distancia Dep<sub>1</sub></b>	<b>Distancia Dep<sub>2</sub></b>	<b>Demanda</b>
1	1	1	1	7	8
2	3	3	3	5	8
3	4	2	3	3	6
4	2	6	5	7	12
5	7	4	2	6	8
6	9	1	3	7	10
7	10	5	4	9	13
8	6	2	3	4	7
Dep <sub>1</sub>	2	1	0	0	0
Dep <sub>2</sub>	8	3	0	0	0

En la figura 6, se presenta la dispersión de puntos (coordenadas x , y), de clientes y depósitos.



- Algoritmo de Asignación:

Aplicamos el algoritmo de asignación mediante el método húngaro, para generar un individuo acorde con la codificación, mediante la asignación se busca asignar los clientes a los depósitos y luego llamar al algoritmo genético, para ello tomamos la tabla 1 con los datos iniciales del problema, para luego reducir los renglones, tomando el mínimo valor de cada renglón y lo restamos a cada uno de los datos de la tabla, aclarando que este proceso se hace renglón por renglón y columna por columna.

La tabla 6 muestra los datos de las distancias entre cada sitio y entre depósitos.

**Tabla 6**  
**Datos de Distancia**

Dep/Sitio	1	2	3	4	5	6	7	8
1	1	3	3	5	2	3	4	3
2	7	5	3	7	6	7	9	4
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0

Como se mencionó anteriormente, primero seleccionamos el menor valor de cada renglón, y lo restamos a cada valor de las demandas de cada una de las filas.

Para el renglón 1 el menor valor es 1 y para el renglón 2 es 3, restando estos valores obtenemos la tabla 7, la cual contiene los resultados de la tabla original, pero reduciendo los dos primeros renglones.

**Tabla 7**  
**Reducción de renglones 1 y 2**

Dep/Sitio	1	2	3	4	5	6	7	8
1	0	2	2	4	1	2	3	2
2	4	2	0	4	3	4	6	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0

Como se han creado valores ficticios, entonces no es necesario hacer la reducción por columnas puesto que el menor valor sería cero y por tanto los valores serían los mismos.

Después de esto se procede a cancelar con la menor cantidad de líneas posibles los ceros de la tabla, y con ello seleccionar el menor valor no tachado y con él se realiza la resta a cada número no tachado, y con aquellos valores que están en intersección de líneas se le sumará dicho número, con aquellas demandas que están cubiertas por una sola línea, se les deja igual.

Estos resultados se consignan en la tabla 8.

**Tabla 8**  
**Cancelación de Ceros**

Dep/Sitio	1	2	3	4	5	6	7	8
1	0	2	2	4	1	2	3	2
2	4	2	0	4	3	4	6	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0

De esta tabla, seleccionamos el menor valor no tachado, para nuestro ejemplo es el 1.

Con este número debemos restárselo a cada valor que no es tocado por ninguna línea y sumarlo a aquellos valores que están en la intersección de dos líneas.

La tabla 9 contiene los resultados de dichas operaciones.

**Tabla 9**  
**Suma y resta del menor valor no tachado**

Dep/Sitio	1	2	3	4	5	6	7	8
1	0	1	2	3	0	1	2	1
2	4	1	0	3	2	3	5	0
1	1	0	1	0	0	0	0	0
2	1	0	1	0	0	0	0	0
1	1	0	1	0	0	0	0	0
2	1	0	1	0	0	0	0	0
1	1	0	1	0	0	0	0	0
2	1	0	1	0	0	0	0	0

Luego se tendría la siguiente asignación, al depósito 1 le corresponderían aquellos sitios con demanda cero. Es decir: 1, 3, 2, 4 y al depósito 2. Serían 5, 6, 8, 7, con esto se obtiene el individuo  $J_1$ :

$J_1 = [1 \ 3 \ 2 \ 4 \ 5 \ 6 \ 8 \ 7]$  y su respectiva demanda es  $D_1 = [8 \ 6 \ 8 \ 12 \ 8 \ 10 \ 7 \ 13]$  Depo<sub>1</sub> = [4 4]

$V_1 = \begin{bmatrix} 1 & 3 & 0 \\ 1 & 3 & 0 \end{bmatrix}$ ; La matriz  $V_1$ , se obtiene al aplicar el ruteo al individuo  $J_1$ . Sus resultados son presentados en las tablas 10 y 11:

**Tabla 10**

**Ruteo para el depósito 1**

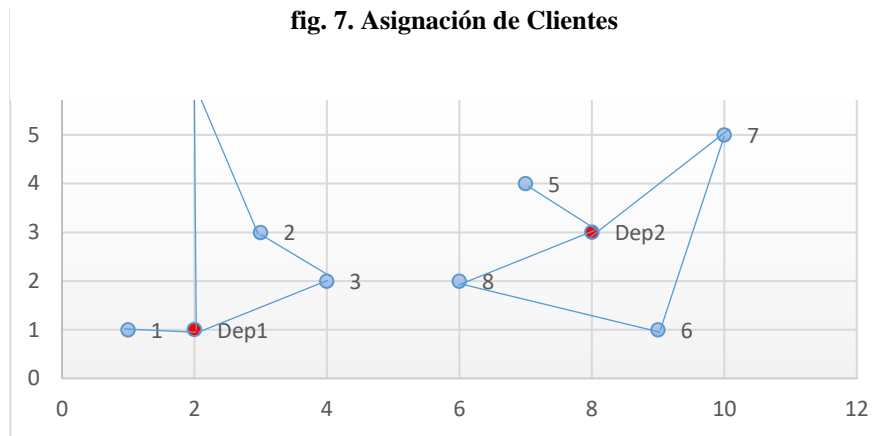
	nodo	dep <sub>1</sub>	1	3	2	4	dep <sub>1</sub>
	demanda	34	8	6	8	12	
Depo <sub>1</sub>	recorrido	15,46	1	2,23	2,23	5	5
	ventana de tiempo	34	7	10	6	5	6

**Tabla 11**

**Ruteo para el depósito 1**

Depo <sub>2</sub>	nodo	dep <sub>2</sub>	5	6	8	7	dep <sub>2</sub>
	demanda	38	8	10	7	13	
	recorrido	25,1	1,41	3,6	5	12,5	4
	ventana de tiempo	34	6	5	10	7	6

La figura 7. Representa la asignación de los clientes a los depósitos y su respectivo ruteo.



**Fo= 24.**

### 6.1 Ruteo: Algoritmo Genético de Chu – Beasley modificado (AGCB)

Mediante la asignación se pueden establecer los datos para cada depósito para así poder generar el ruteo, esto se hace con el vector  $J_n$ . Por ejemplo, para  $J_1 = [1\ 3\ 2\ 4\ 5\ 6\ 8\ 7]$ , se pueden obtener dos sub – vectores  $J_{11}$  y  $J_{12}$ , separando para cada uno de los depósitos de la siguiente manera. Para el depósito 1:  $J_{11} = [1\ 3\ 2\ 4]$ , su respectivo vector demanda es  $D_{11} = [8\ 6\ 8\ 12]$ .

Para el depósito 2:  $J_{12} = [5\ 6\ 8\ 7]$ , su respectivo vector demanda es  $D_{12} = [8\ 10\ 7\ 13]$ ; es decir cada uno de los depósitos recibe un número de clientes (sitios turísticos). Para  $J_1$ , tenemos que están asignados 4 clientes al depósito 1 y cuatro al depósito 2. Los cuales se representan en el vector  $Depo_1 = [4\ 4]$ . Luego se puede aplicar la técnica de ahorros para encontrar una mejor solución de ruteo para cada uno de los depósitos y se obtiene una de las permutaciones para cada uno de ellos:

Para el depósito 1:  $J_{11} = [1\ 4\ 2\ 3]$ ,  $D_{11} = [8\ 12\ 8\ 6]$ ,  $Depo_1 = [4\ 0]$ ,  $V_{11} = [1\ 3]$

Para el depósito 2:  $J_{12} = [5\ 6\ 8\ 7]$ ,  $D_{12} = [6\ 10\ 7\ 13]$ ,  $Depo_1 = [4\ 0]$ ,  $V_{12} = [1\ 3]$ .

Para la población inicial, el AGCB requiere entrar individuos actualizados, para ello, se toman los individuos del algoritmo genético y se permutan aplicando la técnica de ahorros, la cual consiste en hallar las distancias entre cada nodo, y luego hallar el mayor ahorro para con ello buscar una permutación de la solución que mejore la función objetivo, después, con la ayuda de algoritmos

heurísticos se generará una población de individuos diferentes, para luego utilizar el AGCB y encontrar la mejor solución a cada sección del individuo A.

A continuación, se presenta un ejemplo para actualizar el individuo  $J_{11}$  y  $J_{12}$  para considerarlo como una posible solución.

Para  $J_{11}$ , tenemos que  $J_{11} = [1\ 4\ 2\ 3]$

$D_{11} = [8\ 12\ 8\ 6]$

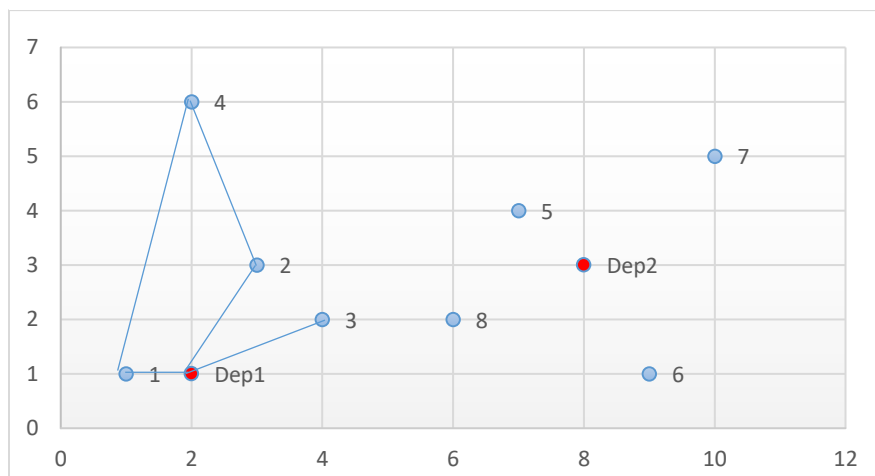
$Depo_1 = [4\ 0]$

$V_{11} = [3\ 1]$

$Fo = 18$

En la figura 8, se muestra la actualización del individuo  $J_{11}$

**fig. 8. Actualización de  $J_{11}$**



Para  $J_{12}$ , tenemos que  $J_{12} = [5\ 6\ 8\ 7]$

$D_{12} = [8\ 10\ 7\ 13]$

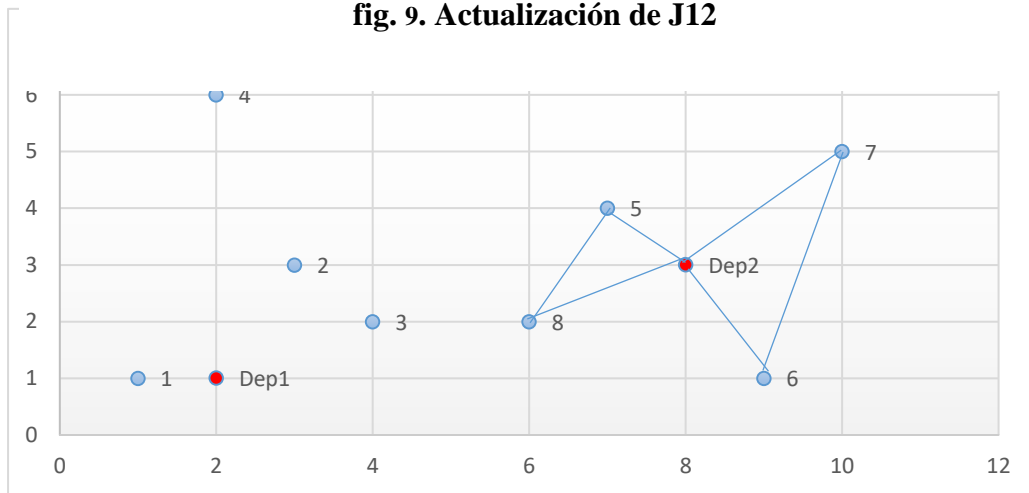
$Depo_1 = [4\ 0]$

$V_{12} = [2\ 2]$ .

$Fo = 17,3$

La figura 9, representa la actualización del individuo  $J_{12}$ .



**fig. 9. Actualización de J12**

En el AGCB, se necesita integrar nuevamente los clientes de cada depósito, es decir nuevamente retomamos el individuo  $J_1$ , es decir, tenemos:

$$J_1 = [1 \ 3 \ 2 \ 4 \ 5 \ 6 \ 8 \ 7]; D_1 = [8 \ 6 \ 8 \ 12 \ 8 \ 10 \ 7 \ 13]; Depo_1 = [4 \ 4]; V_1 = \begin{bmatrix} 1 & 3 & 0 \\ 1 & 3 & 0 \end{bmatrix}; Fo = 35,3$$

Luego, mediante los algoritmos de mejoramiento, se procede, a hacer los intercambios entre rutas y depósitos, para ello usamos los algoritmos shift (1,0) y (2,0).

Aplicamos shift (1,0) al individuo  $J_1$ , para obtener un nuevo individuo  $J_3$ . Al iterar el algoritmo se obtiene:

$$J_3 = [1 \ 3 \ 2 \ 4 \ 5 \ 6 \ 7 \ 8]$$

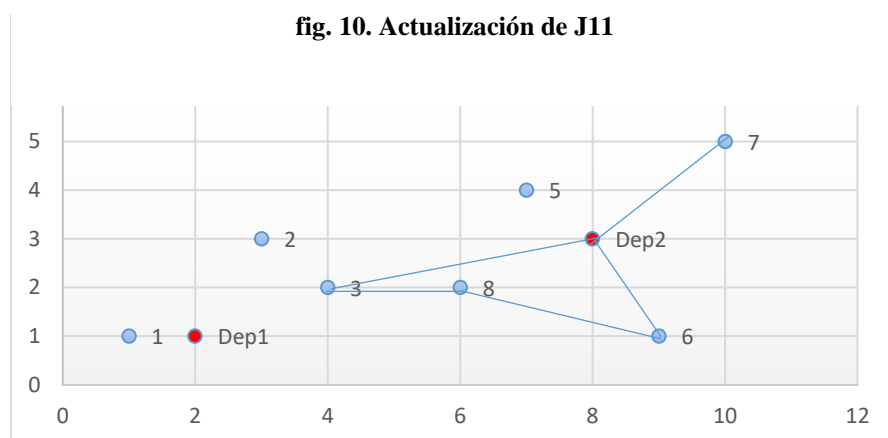
$D_3 = [8 \ 6 \ 8 \ 12 \ 8 \ 10 \ 13 \ 7]$ ; aplicando nuevamente el ruteo como se muestra en las tablas 6 y 7 se obtiene:

$$Depo_3 = [4 \ 4]$$

$$V_3 = \begin{bmatrix} 3 & 1 & 0 \\ 3 & 1 & 0 \end{bmatrix}$$

$Fo = 32,3$  en este caso hay mejora.

La figura 10, representa el intercambio entre clientes, resultado que se obtiene de la aplicación del shift (1,0) al individuo  $J_1$ .



## 6.2 Desarrollo metodológico

### 6.2.1 Asignación de clientes a los depósitos:

El asignar se realiza de los clientes a los depósitos, usando el modelo de asignación con ayuda del método húngaro, en el cual se debe fijar  $n$  objetos o personas invisibles a  $m$  tareas de una manera óptima, esto se muestra en el ejemplo anterior.

Un problema debe cumplir las siguientes propiedades para formularse como un problema de asignación según Chávez en [15]:

El número de objetos o personas es el mismo número de tareas.

A cada persona se le asigna una tarea.

Existe un costo  $C_{ij}$  de asignación de la persona  $i$  a la tarea  $j$ .

El objetivo es buscar la combinación que minimice los costos totales.

### 6.2.2.1 Modelo de asignación

Las variables utilizadas en el modelo de asignación son variables binarias, es decir, variables que pueden tomar solo dos valores 0 y 1.

Matemáticamente:  $x_{ij} = \begin{cases} 1, & \text{si el asignado } i \text{ realiza la tarea } j \\ 0, & \text{en caso contrario} \end{cases}$

Para  $i = 1, 2, 3, \dots, n$  y  $j = 1, 2, 3, \dots, n$

El costo total de la asignación es igual a la suma de los productos de cada variable  $x_{ij}$  por el costo asignado  $C_{ij}$ .

$$Z_{min} = \sum_{i=1}^n \sum_{j=1}^n X_{ij} C_{ij} \quad 4.1$$

En las restricciones se asigna a una persona a cada una de las tareas y cada tarea debe ser realizada por una persona.

$$\sum_{j=1}^n X_{ij} = 1 \text{ para } i = 1, 2, 3, \dots, n \quad 4.2$$

$$\sum_{i=1}^n X_{ij} = 1 \text{ para } j = 1, 2, 3, \dots, n \quad 4.3$$

El modelo completo de asignación se obtiene al añadir la restricción de no negatividad y la de variables binarias.

$$Z_{min} = \sum_{i=1}^n \sum_{j=1}^n X_{ij} C_{ij} \quad 4.4$$

Sujeto a:

$$\sum_{j=1}^n X_{ij} = 1 \text{ para } i = 1, 2, 3, \dots, n \quad 4.5$$

$$\sum_{i=1}^n X_{ij} = 1 \text{ para } j = 1, 2, 3, \dots, n \quad 4.6$$

$$X_{ij} \text{ binarias } \forall i, j$$

$$X_{ij} \geq 0$$

En este sentido, se observa que el modelo de asignación es muy parecido al de transporte, la diferencia radica en que las variables del modelo de asignación son binarias, mientras que en el de transporte son enteras.

Entonces se puede tomar el modelo de asignación como un problema de transporte donde cada una de las personas es el origen y cada una de las tareas son los destinos. La oferta y demanda son igual a uno, es decir, cada origen tiene una sola persona y cada destino necesita sólo una persona. Los costos de capacitación representan el costo de transportar una unidad del origen  $i$  al destino  $j$ . Por lo tanto, el objetivo es encontrar la combinación que minimice los costos de asignación y cumpliendo las restricciones de oferta y demanda [15].

### 6.2.2 Técnicas para ruteo:

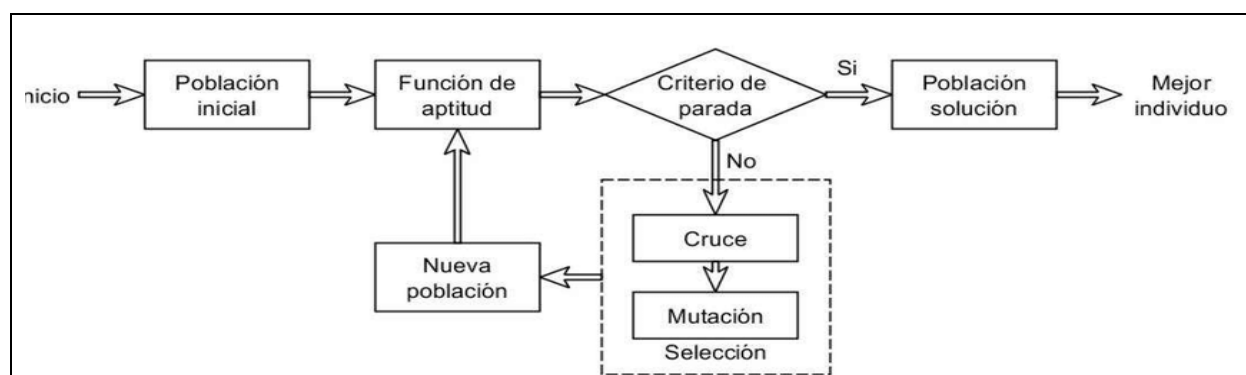


fig. 11. Diagrama de flujo para ruteo

Tomado de: [https://www.researchgate.net/figure/Figura-1-Diagrama-de-flujo-de-un-algoritmo-genetico-simple\\_fig1\\_310731033](https://www.researchgate.net/figure/Figura-1-Diagrama-de-flujo-de-un-algoritmo-genetico-simple_fig1_310731033)

Posteriormente, terminada la primera etapa se cuenta con la población inicial, para con ella, en la segunda etapa resolver el VRPTW para determinado depósito generando una o más rutas para cada uno de ellos, es decir, se hacen las iteraciones de forma independiente aplicando el AGS a cada uno de los tres depósitos, luego haciendo uso del algoritmo genético de Chu- Beasley (AGCB) modificado y las heurísticas se configura una población inicial, en donde la información del individuo permite establecer las mejores soluciones y así tener los datos para cada depósito para finalmente generar el ruteo [1].

En la tercera etapa se realiza el mejoramiento, es decir, a la solución encontrada se aplican algoritmos de intercambios de aristas y nodos mejorando la solución, para ello usaremos los algoritmos de mejoramiento 2-opt y shift (1,0) y (2,0), seleccionando finalmente la mejor solución obtenida y finalmente mostrándola en forma gráfica.

El diagrama de flujo mostrado en la figura 11, se explica a continuación.

#### **6.2.2.1 Algoritmo genético:**

Según Bermeo en [20], los algoritmos genéticos son métodos sistemáticos para la resolución de problemas de búsqueda y optimización que aplican a estos, los mismos métodos de la evolución biológica: selección basada en la población, reproducción sexual y mutación. En un algoritmo genético, tras parametrizar el problema en una serie de variables,  $x_1, \dots, x_n$  se codifican en un cromosoma. Las soluciones codificadas en un cromosoma compiten para ver cuál constituye la mejor solución (aunque no necesariamente la mejor de todas las soluciones posibles).

Una vez asignados los clientes a los diferentes depósitos, se procede a la segunda etapa que es rutear determinado depósito, en cada uno de ellos se corre un AG, y se genera una o más rutas según la demanda del cliente (Dinero requerido en cada sitio turístico) y la capacidad del vehículo (Dinero del turista), usando el algoritmo genético modificado de Chu-Beasley.

De este modo, el proceso algorítmico se realiza mediante los siguientes pasos:

- 1. Configuración del MDVPRTW:** Consiste en tomar los datos correspondientes al problema, es decir los clientes con sus coordenadas y sus demandas, el número de depósitos y las ventanas de tiempo.
- 2. Selección de padres:** Se genera una población inicial de 30 soluciones, esto ya que el problema es relativamente pequeño (menos de 50 sitios turísticos) a través del algoritmo genético simple y las heurísticas de ahorro, shift entre otras, tal como se explica en el ejemplo anterior, es decir, primero se asignan los clientes a los depósitos con ayuda del

algoritmo de asignación, después se actualizan los individuos y con ayuda de las heurísticas de ahorros se procede a permutar los individuos para ir mejorando cada uno de ellos, y estos son actualizados mediante la aplicación de las técnicas shift (1,0) y (2,0) y luego se obtiene la primera generación seleccionando los dos primeros padres por torneo, es decir, aleatoriamente se toman cuatro padres y de ellos seleccionamos los dos mejores, éstos serán los de menor costo, es decir los de mejor función objetivo, teniendo en cuenta este esquema se tiene garantía de que si el peor de los padres es seleccionado para el torneo, no lo podrá ganar y por tanto no podrá generar hijos, es decir solo los padres buenos tendrán hijos.

3. **Cruce:** luego de seleccionados los dos padres mejores, se realiza el cruce entre ellos, utilizando el emparejamiento parcial, en el cual se elige uno o dos subsegmentos de uno de los padres y cruzarlos conservando el orden y la posición de la mayor cantidad de genes posibles del otro manteniendo la coherencia, para esto se realiza de la siguiente manera:
  - A. Un punto de corte aleatorio, (un subsegmento) en este caso, se procede a cruzar los hijos según el punto de corte, luego de ello se calcula la función objetivo de cada nuevo hijo y se comparan los valores de esta, si mejora entonces este entra a la población, de lo contrario no. Esto se muestra con detalle en el ejemplo 2.
  - B. Con dos puntos de corte aleatorios, (dos subsegmentos) se combinan los padres iniciales y se sacan varios hijos para ser comparados según el resultado de la función objetivo, en este caso, ingresa a la población inicial siempre y cuando sea mejor y no esté repetido.
4. **Mutación:** se hace la mutación al 5% de las generaciones las cuales se obtienen según la ecuación:  $gener = m * clientes + b$ , como lo indica Guasmayan en la tesis [1], en donde se la realiza por sintonización en CVRP y se valida con instancias de AUGERAT. Es decir, al tomar un número aleatorio debe ser menor que 0,05, luego se escoge uno de los  $n$  hijos aleatoriamente, cuando sea seleccionado se escoge también de forma aleatoria un gen y se le pregunta al algoritmo si se va a mutar o no cogiendo un número aleatorio y comprobando si es menor que el 5%, en este proceso no hay nada controlado todo es aleatorio.

- 5. Mejoramiento:** se realiza con algoritmos de intercambio de rutas entre depósitos y de clientes entre rutas, con la utilización de los algoritmos 2-opt y shift (1,0) y (2,0), tal como se mostró en el ejemplo anterior.

En términos generales se tiene un algoritmo de asignación como generador de soluciones de inicio (Población inicial) que junto con las heurísticas permiten obtener individuos solución mejorados como población inicial, esto mediante la aplicación del algoritmo genético de Chu Beasley, el cual tiene como característica principal garantizar que los individuos sean diferentes y cada vez mejores. Para esto:

Se genera una población inicial con la aplicación del algoritmo de asignación, y a esta población se aplica el algoritmo genético modificado de Chu-Beasley y las heurísticas *2-opt* y *shift (1,0)* y *(2,0)*, para con ello poder reemplazar los peores individuos por nuevos que sean mejores y diferentes.

Los individuos generados, contienen el orden de las visitas, la partición de rutas y el valor de su función objetivo.

Verificar cumplimiento de iteraciones del algoritmo genético de Chu-Beasley.

Realizar intercambio 2-OPT en cada ruta y dar la respuesta final del algoritmo genético de Chu Beasley.

### **Ejemplo:**

Los datos para el ejemplo se presentan en la tabla 12, se toman 3 vehículos, la capacidad de 16 y dos depósitos restricción de la ventana de tiempo 58 horas.

**Tabla 12**  
**Datos de entrada ejemplo**

x	y	demanda	ventanas de tiempo en horas	Nodo	dep <sub>1</sub>	dep <sub>2</sub>	1	2	3	4	5	6	7	8
2	1	18	6	dep <sub>1</sub>	0	6,32	1	2,23	2,23	5	5,83	7	40	8,5
8	3	18	5	dep <sub>2</sub>	6,32	0	7,28	5	4,12	6,7	1,41	2,23	4	2,5
1	1	4	7	1	1	7,28	0	2,82	3,16	5,09	6,7	8	48,5	13
3	3	2	10	2	2,23	5	2,82	0	1,41	3,16	4,12	6,32	26,5	5
4	2	3	6	3	2,23	4,12	3,16	1,41	0	4,47	3,6	5,09	22,5	2
2	6	1	5	4	5	6,7	5,09	3,16	4,47	0	5,38	8,6	32,5	16
7	4	3	7	5	5,83	1,41	6,7	4,12	3,6	5,38	0	3,6	5	2,5
9	1	5	8	6	7	2,23	8	6,32	5,09	8,6	3,6	0	8,5	5
10	5	6	12	7	40	4	48,5	26,5	22,5	32,5	5	8,5	0	12,5
6	2	6	8	8	8,5	2,5	13	5	2	16	2,5	5	12,5	0

Después de asignar y rutear, tal como se muestra en el ejemplo anterior. Se obtienen las tablas 13 y 14, en las cuales se consignan las rutas generadas para cada depósito, la demanda de cada una de ellas y el vector de ventanas de tiempo.

En ninguna de las rutas se violan las restricciones por tanto se consideran buenas soluciones

**Tabla 13**

**Ruteo depósito 1**

Ruta 1	Ruta	dep <sub>1</sub>	1	3	2	4	dep <sub>1</sub>
	demanda	15	4	8	2	1	
	recorrido	15,46	1	2,23	2,23	5	5
	ventana de tiempo	34	7	10	6	5	6

**Tabla 14**

**Ruteo depósito 2**

Ruta 2	Ruta	dep <sub>2</sub>	5	6	8	7	dep <sub>2</sub>
	demanda	16	1	3	6	6	
	recorrido	22,6	1,41	3,6	2,5	12,5	4
	ventana de tiempo	34	6	5	10	7	6

Luego aleatoriamente se seleccionan 4 padres y se toman a los dos mejores, quedando seleccionados P1 y P2, en donde:



$P_1 = [1\ 3\ 2\ 4\ 5\ 6\ 8\ 7]$  y  $P_2 = [5\ 6\ 8\ 7\ 3\ 2\ 4\ 1]$ , Calculamos sus respectivas funciones objetivo de cada padre obteniendo los siguientes resultados:

$FO_{P_1} = 75,2$  y  $FO_{P_2} = 39,4$ , después de esto tomamos un punto de corte aleatorio para realizar el cruce entre padres, por ejemplo, con  $rand(i) = 3$ , tendríamos un punto de corte aleatorio entre el tercer y cuarto nodo de cada uno de ellos, o sea:  $P_1 = [1\ 3\ 2\ | \ 4\ 5\ 6\ 8\ 7]$  y  $P_2 = [5\ 6\ 8\ | \ 7\ 3\ 2\ 4\ 1]$ , con ello se procede a generar los hijos, cruzando los genes de uno de los padres con los del otro, tal como se muestra en los colores, quedando los hijos de la siguiente manera.

Hijo uno,  $H_1 = [1\ 3\ 2\ 7\ 3\ 2\ 4\ 1]$ , aquí se le da prioridad al padre 1 para realizar la mutación.

Hijo dos,  $H_2 = [5\ 6\ 8\ 4\ 5\ 6\ 8\ 7]$  en este hijo tendrá relevancia el padre 2 para la mutación.

Como se ve en cada uno de los hijos hay genes que se repiten, por tanto, se hace necesario realizar la mutación, para ello simplemente se mutan o remplazan los genes repetidos por los que falten dando prioridad a cada padre, los valores a mutar en el hijo 1 están resaltados en color verde y para el hijo 2 se resaltan en color gris.

Luego de realizar la mutación tenemos los nuevos hijos.

$H_3 = [1\ 3\ 2\ 7\ 5\ 6\ 4\ 8]$  y  $H_4 = [5\ 6\ 8\ 4\ 3\ 2\ 1\ 7]$  en donde, sus respectivas funciones objetivos son:

$FO_{H_3} = 73,8$  y  $FO_{H_4} = 87,2$ .

Entonces, como  $FO_{H_3} = 73,8 < FO_{P_1} = 75,2$ , se debe reemplazar el  $P_1$  por el  $H_3$ , pero antes de ello se debe preguntar si éste vector está en la población, si está no se reemplaza.

Suponiendo que no este, lo reemplazamos y procedemos a aplicar el algoritmo genético para hacer el ruteo a  $H_3$ , teniendo los siguientes resultados:

$H_3 = [1\ 3\ 2\ 7\ 5\ 6\ 4\ 8]$ ;

La demanda es  $DH_3 = [4\ 3\ 2\ 6\ 3\ 5\ 1\ 6]$ ;

Su respectivo vector de ventanas de tiempo es  $TW = [7\ 6\ 10\ 12\ 7\ 8\ 5\ 8]$ ;

Con estos datos tenemos que el primer vehículo alcanza a recorrer los 4 primeros nodos, el segundo vehículo los siguientes 4 y el tercer vehículo ninguno, recordando que la capacidad máxima es 16.

Tenemos entonces que  $V = [4\ 4\ 0]$ ;

Los respectivos valores para las ventanas de tiempo (TW) de cada vehículo son:

Vehículo 1;  $TW_1 = [7\ 6\ 10\ 12] = [35]$

Vehículo 2:  $TW_2 = [7 \ 8 \ 5 \ 8] = [28]$

Vehículo 3:  $TW_3 = [0] = [0]$ .

Finalmente, se calcula la función objetivo para el  $H_3$ ,

$FO H_3 = [Depo_1 \ 1 \ 3 \ 2 \ 7 \ Depo_1] + [Depo_1 \ 5 \ 6 \ 4 \ 8 \ Depo_1] = 1 + 3,16 + 1,41 + 26,5 + 40 + 5,83 + 3.6 + 8,6 + 16 + 8,5 = 114,6$ .

### ***6.2.3 Intensificación algoritmo genético usando Intercambios entre vecinos cercanos.***

Tanto para la mutación como para el mejoramiento se procede a realizar una pesquisa por medio del algoritmo de búsqueda en vecindario o entorno variable denominada VNS que consiste en hacer una exploración de mejoras sobre los óptimos locales.

Para el caso de mejoramiento de la solución alcanzada por el algoritmo genético se realiza el movimiento intercambiando vecinos del vector de rutas y verificando si los cambios mejoran la solución y teniendo en cuenta la capacidad de los vehículos, como se indica en [17] [1].

## **VII. CARACTERIZACIÓN DEL CASO DE PRUEBA PARA LOS RECORRIDOS TURÍSTICOS EN EL DEPARTAMENTO DE NARIÑO.**

El departamento de Nariño cuenta con varios sitios turísticos, de los cuales se han seleccionado los más representativos y más llamativos, éstos son visitados por distintos turistas que arriban al departamento por medio de tres diferentes ciudades, en variados medios de transporte, tal es el caso del municipio de Chachagüí, al cual pueden acceder por vía aérea mediante al aeropuerto Antonio Nariño.

Ahora bien, otra ciudad de arribo de pasajeros es la capital San Juan de Pasto, a la cual se accede por medio del terminal terrestre de pasajeros, y el último punto de acceso es la ciudad fronteriza de Ipiales mediante el terminal de transporte terrestre y también se tiene acceso por el aeropuerto de San Luis, el cual se desconoce su pertenencia a dicho municipio o a un municipio vecino, por esta razón, tomaremos como referencia de acceso a esta ciudad el terminal terrestre de pasajeros.

De acuerdo con lo anterior, se toma cada uno de estos puntos de acceso al departamento como los depósitos y con base en ellos se generan las diferentes rutas.



**fig. 12. Aeropuerto Antonio Nariño Chachagüí.**

Fuente: <https://www.aviacol.net>. Portal de aviación.



**fig. 13. Depósito 2. Terminal Terrestre de Pasajeros Pasto.**

Fuente: <https://www.horariodebuses.com.co/>.



**fig. 14. Depósito 3. Terminal Transporte Terrestre Ipiales.**

Fuente: <https://www.horariodebuses.com.co/>.

### ***7.1 Descripción de las rutas turísticas.***

Según entrevista realizada a funcionarios de la secretaria de turismo departamental, Nariño tiene tres rutas turísticas principales trazadas por zonas, la zona sur, la zona centro y la zona norte, y dentro de éstas se destacan sitios que permiten realizar diferentes clases de turismo entre los cuales

se destacan el turismo religioso, histórico, café, sol y playa, turismo primordial, turismo natural, y un turismo Cultural. De estas zonas las más visitadas son las dos primeras.

Con base en esta información junto con las ciudades de acceso al departamento se toman tres de ellas como depósitos y a partir de ellos se generan rutas óptimas para visitar el departamento de Nariño.

Los sitios turísticos definidos en el presente trabajo son tomados según la clase de turismo que puede hacerse en el departamento, es decir se puede ofrecer distintas actividades a realizar dentro del recorrido por las diferentes rutas, tales como, actividades religiosas, de aventura, deportes extremos, descubrimiento de especies, recorridos por y avistamiento de flora y fauna, entre otras. Con estas características se tomaron como referencia los 33 puntos que se podrán conocer de forma óptima con ayuda de la presente investigación además de permitir al departamento surgir y generar nuevos ingresos fuera de la actividad principal que es la agricultura.

Se aclara que para el presente trabajo se considera el ruteo de vehículos agrupando los clientes cercanos en sectores (zonas), que corresponden a las regiones norte sur y centro del departamento, definiendo para ello los datos de latitud y longitud de cada sitio turístico presentados en la tabla 15.

**Tabla 15**  
**Coordenadas latitud longitud**

CIUDAD	No.	Coordenada latitud	Coordenada Longitud
CUIDAD DE PASTO	3	1,205784	-77,285759
LAS LAJAS	4	0.805816,	-77.586086
VOLCAN CUMBAL	5	0,864306	-77,861121
VOLCAN CHILES	6	0,864306	-77,861121
RICAURTE LA PLANADA	7	1,212308	-78,002041
LA COCHA	8	1,103739	-77,155042
LAS MESAS	9	1,475793	-77,020579
LAGUNA DE TELPIS	10	1,125732	-77.399.905
TUQUERRES	11	1,085654	-77,618684

LAS MESAS	12	1,475793	-77,020579
TERMALES DE TAJUMBINA	13	1,599462	-77,131361
RICAUORTE	14	1,224238	-77,996351
SANDONA	15	1,283131	-77,471401
SAN LORENZO	16	1,503477	-77,214588
BUESACO	17	1,382839	-77,155805
TUMACO	18	1,787434	-78,790848
SAN PABLO	19	1,669068	-77,012597
EL TAMBO	20	1,408035	-77,391528
BOMBONA	21	1,196494	-77,464233
BARBACOAS	22	1,668251	-78,134563
CUMBITARA	23	1,686658	-77,571969
FUNES	24	1,001039	-77,448926
CASCADA DEL WILKE	25	1,205784	-77,285759
EL PEDREGAL	26	0,825235	-77,625273
TAMINANGO	27	1,570372	-77,280813
MALLAMA	28	1,148227	-77,882766
GUAITARILLA	29	1,130499	-77,550093
LA UNION	30	1,599462	-77,131361
GUALMATAN	31	0,919366	-77,566546
SAMANIEGO	32	1,33257	-77,585729
LINARES	33	1,350804	-77,523366
PASTO terminal	1	1,19835	-77,269342
IPIALES Terminal	2	0,85866	-77,675652
CHACHUAGUI	22	1,397253	-77,290516

Gráficamente se presenta una ruta turística, la cual se define como la circunvalar al Galeras, figura 15, utilizando para ello la aplicación de Google Maps, dicha ruta tiene una distancia de 210 km, que permite llegar hasta el cráter del Volcán Galeras y visitar la imponente naturaleza que lo rodea,

por medio de ésta se puede conocer el santuario de flora y fauna nariñense que abastece de agua a varios municipios del departamento.



## 7.2 Demanda de los productos por clientes agrupados en sectores

El valor de la demanda se obtiene para cada sitio, teniendo en cuenta el costo en el precio de hotel, comida, pasajes internos, número de personas y el número de días que se necesitan para visitarlo y conocerlo, esta información no existe en ninguna base de datos de la oficina de turismo por tanto se hace necesario para muchos de ellos ir personalmente hasta el municipio correspondiente o hasta el más cercano para obtener dicha información y en otros casos conseguirla por medio de llamadas telefónicas.

Luego, se obtiene un valor en pesos (\$) correspondiente a la demanda total, para ello agrupamos por sectores (zonas) cada una de las regiones del departamento, y considerando el valor de cada sitio se obtiene dicha demanda total, teniendo en cuenta que los depósitos como son también sitios turísticos entonces requieren de un presupuesto para ser visitados, por lo tanto, si son considerados dentro de los sectores (zonas), lo mencionado anteriormente se puede apreciar más detalladamente

en la siguiente tabla 7, la cual resume en su valor de demanda los datos obtenidos del precio de hotel, comida y transportes internos, resaltando que los datos obtenidos fueron determinados desde 1 día, hasta un promedio de 7 días, que son tiempos estimados necesarios para conocer cada uno de los sitios del departamento.

Se anexa en la tabla 16, los datos obtenidos para encontrar la demanda de cada uno de los sitios turísticos.

**Tabla 16**  
**Matriz de Demandas**

CIUDAD	DEMANDA (\$)
CIUDAD DE PASTO	256000
LAS LAJAS	150000
VOLCAN CUMBAL	164000
VOLCAN CHILES	294000
RICAURTE	255000
LA COCHA	276000
LAGUNA DEL SILENCIO	480000
LAGUNA DE TELPIS	40000
TUQUERRES	132000
COMPLEJO VOLCÁNICO DOÑA JUANA	480000
TERMALES DE TAJUMBINA	450000
RICAURTE LA PLANADA	270000
SANDONA	40000
SAN LORENZO	88000
BUESACO	156000
TUMACO	2310000
SAN PABLO	41000
EL TAMBO	33000
BOMBONA	46000

BARBACOAS	136000
CUMBITARA	240000
FUNES	73000
CASCADA DEL WILKE	268000
EL PEDREGAL	218000
TAMINANGO	200000
MALLAMA	90000
GUAITARILLA	60000
LA UNION	435000
GUALMATAN	130000
SAMANIEGO	126000
LINARES	166000
PASTO terminal	444000
IPIALES	258000
CHACHUAGUI	348000

### ***7.3 Descripción de la matriz de costos y distancias***

Con los datos de la Tabla 6 sobre longitud y latitud de cada sector turístico, obtenidos con la ayuda de Google Maps, se calcula el valor de la distancia real desde cada depósito, teniendo en cuenta que para algunos sitios se hace necesario aproximar estos valores, puesto que el acceso al mismo no registra el programa. (Cada sitio turístico se identifica como cliente 1, cliente 2, ...)

Según la información anterior se define para el caso de prueba de 33 sitios incluidos los tres depósitos, especificando la matriz de costos de 33 filas por 33 columnas. Como se muestra en la tabla 17.



**Tabla 17****Matriz de costos. 31x31**

	Cliente1	Cliente 2	Cliente 3	...	Cliente 32	Cliente 33	Dep1	Dep2	Dep3
<b>Cliente 1</b>	0	90,2	10,8	...	117	81,8	33,1		
<b>Cliente 2</b>	90,2	0	101	...	107	172			
<b>Cliente 3</b>	10,8	101	0	...	128	90,7			
<b>...</b>	...	...	...	...	...	...			
<b>Cliente32</b>	117	107	128	...	0	28			
<b>Cliente33</b>	81,8	172	90,7	...	28	0			
<b>Dep1</b>							0		
<b>Dep2</b>							33,1	0	
<b>Dep3</b>							117		0

**7.4 Vector de ventanas de tiempo**

El vector de ventanas de tiempo se define con los tiempos promedio en días que se requieren para visitar, conocer y recorrer cada uno de los sitios turísticos, teniendo en cuenta los tiempos que se necesitan para llegar hasta cada uno.

El vector que contiene estos datos se muestra en la tabla 18.

**Tabla 18****Vector de las ventanas de tiempo.**

CIUDAD	#DÍAS
PASTO	3
LAS LAJAS	2
PASTO	2
VOLCAN CUMBAL	2
VOLCAN CHILES	3

RICAURTE LA PLANADA	3
LA COCHA	2
LAS MESAS	2
LAGUNA DE TELPIS	1
TUQUERRES	2
LAS MESAS	2
UNION	2
RICAURTE	2
SANDONA	1
SAN LORENZO	1
BUESACO	2
TUMACO	7
SAN PABLO	1
EL TAMBO	1
BOMBONA	1
BARBACOAS	1
CHACHUAGUI	3
CUMBITARA	2
FUNES	1
PASTO	2
EL PEDREGAL	2
TAMINANGO	2
MALLAMA	1
GUAITARILLA	1
LA UNION	3
GUALMATAN	2
SAMANIEGO	3
LINARES	2

Los costos se consideran todos en la misma unidad en este caso pesos (\$), ya que todas las variables se pueden normalizar con esta unidad, considerando que la distancia se puede evaluar por medio del pasaje interno que se debe pagar para llegar al sitio, en este sentido, el tiempo es sinónimo de la duración del turista en el lugar, lo cual está sujeto a la cantidad de dinero que él disponga.

## VIII. METODOLOGÍA

La presente investigación se lleva a cabo en diferentes fases que permiten dar solución al problema, en una etapa inicial se estudia el problema de ruteo de vehículos, y su variación multi depósitos con ventanas de tiempo, así como las técnicas de solución, luego se formula y se implementa la metodología para finalmente validar los resultados obtenidos.

Para el objeto de este trabajo, el método de investigación apropiado a seguir es un modelo descriptivo el cual este enfocado a la aplicación de técnicas de optimización y junto con el desarrollo de la metodología de solución para el problema del *ruteo de vehículos con múltiples depósitos y ventanas de tiempo* se pretende encontrar una solución de calidad, efectiva y sobre todo óptima considerando la minimización del tiempo de procesamiento en los algoritmos de solución y la búsqueda de rutas que minimicen el costo de los viajes de acuerdo con la distancia recorrida por los vehículos, su capacidad, y el número de personas y de días que vaya a quedarse, conformes al horario del recorrido.

Para lograr la solución del modelo se deben realizar varias pruebas y validar con ello la solución encontrada, esto con el propósito de conocer si la metodología desarrollada es adecuada y presenta resultados de buena calidad, (esto se realiza por medio de pruebas en simulación y diferentes corridas con parámetros encontrados en la literatura especializada para diferentes instancias publicadas para el modelo de VRP, MDVRP y MDVRPTW) [1]. Esto permite validar el software que se diseña para la solución del problema.

El desarrollo metodológico está dividido en seis fases, las cuales describen el desarrollo del proyecto por medio de los subitems.

A continuación, se mostrarán las fases y su subdivisión.

### **FASE I: Diagnostico del Problema**

- Contexto del Problema.
- Determinar los Objetivos a alcanzar.

### **FASE II: Identificación de un Modelo Matemático**

- Definir Parámetros y Variables.

Se hace la correlación entre los parámetros del modelo y las variables del problema con el que se plantea en esta investigación:

- Los vehículos serán considerados las personas, en nuestro caso de prueba son los turistas.
  - Depósitos – Municipios a los cuales se puede arribar desde cualquier parte del País al departamento de Nariño, para nuestra investigación son los municipios de San Juan de Pasto, Chachagüí, e Ipiales, con lo cual se convierte en un problema de ruteo de vehículos con múltiples depósitos.
  - Clientes – se los considera como los sitios del departamento a los cuales el turista va a ir a visitar, en cada uno de éstos, el turista va a dejar dinero, y a su vez cada sitio demanda un cierto costo.
- Determinar la Función Objetivo.

El objetivo es minimizar costos, es decir minimizar el valor que implica ir de un lugar a otro, garantizando así que el turista visite más sitios con el dinero que dispone.

- Determinar las restricciones.

- Determinar los algoritmos que faciliten la solución del problema.
- Definir una estrategia matemática de solución.

### **FASE III: Obtención de una Solución a partir del Modelo**

- Adecuar el algoritmo de Chu-Beasley para que solucione el problema en tiempos computacionales convenientes utilizando un lenguaje de programación acertado, de tal manera que pueda ser implementado en una herramienta computacional como MATLAB.
- Obtener resultados del programa parciales

### **FASE IV: Prueba del Modelo de solución.**

- Adicionar diferentes restricciones al problema de ruteo de vehículos.

Examinar de manera exhaustiva que todas restricciones del modelo se estén cumpliendo.

Realizar las corridas necesarias, hasta encontrar coherencia en los planes turísticos de la ruta

### **FASE V: Establecimiento de Controles sobre la Solución**

Determinar efectos de perturbaciones en los algoritmos de solución del problema.

### **FASE VI: Validación de la Metodología de Solución**

Ejecutar la metodología de solución y compararla con la solución obtenida, en los casos de la literatura especializada.

## IX. RESULTADOS

Se prueba el algoritmo implementado generando varias corridas del programa codificado en Matlab, instalado en un equipo portátil con procesador Intel® Core™ i7-6500U CPU @ 2.50GHz 2.60GHz y Memoria (RAM) 4.00 GB.

En primer lugar se establece una calibración de los parámetros propios del problema, posteriormente gracias a los diseños de Cordeau [21] se realizan las corridas con algunos casos de prueba con la creación de redes y optimización emergente, En tercer lugar, se establece un caso de prueba para los sitios turísticos del departamento siguiendo los lineamientos planteados Guasmayan en [1], donde se considera además de la distancia el costo de combustible [17].

Es decir, para garantizar la funcionalidad del algoritmo se realizan pruebas iniciales sobre instancias con resultados definidos, para los cuales se observa la consistencia del algoritmo en sus respuestas según el error relativo y la desviación estándar en las múltiples corridas.

Es decir, con las instancias de Cordeau, se validan los algoritmos genéticos para múltiples depósitos y como ellas también incluyen la restricción de las ventanas de tiempo, entonces se valida el algoritmo para ser aplicado al caso real, ya que la instancia Pr02 tiene 50 clientes, 4 depósitos y 3 vehículos y en ella se cumple el AGCB con ventanas de tiempo, entonces para nuestro caso como son menos clientes y menos depósitos se considera validado el algoritmo.

### ***9.1 Calibración de parámetros:***

Se calibra primero el VRP con instancias de Augerat, luego el MDVRP con Cordeau y aparte se calibraron el número de soluciones, el número de individuos, la mutación, el número de pivotes para selección en torneo, el número de pivotes para cruce, entre otros parámetros, los cuales se hacen corriendo varias veces el algoritmo hasta sintonizarlo, como lo muestra Guasmayan en [1].

Se seleccionan los siguientes valores que deben tomar los parámetros para desarrollar la eficiencia algorítmica.

**Tabla 19**  
**Parámetros del Algoritmo**

<b>PROCESO</b>	<b>PARÁMETRO</b>	<b>DESCRPCIÓN DEL PARÁMETRO</b>
Población inicial	Por torneo	Por medio del algoritmo de asignación se genera la población inicial.
	Porcentaje de soluciones que se envían al algoritmo genético	Elegir las mejores soluciones Realizar intercambios
Evaluación y Generación de la nueva población	Número de generaciones	Valor de las iteraciones del algoritmo genético
	Número de individuos	Número de soluciones diferentes para el ruteo
	Porcentaje de mutación	Porcentaje de probabilidad de que un ser cambie su solución
Algoritmo Genético de Chu-Beasley	Número de individuos seleccionados para torneo	Selección de dos padres con una, dos y tres iteraciones
	Puntos de recombinación	Uno o dos puntos de recombinación de individuos

### **9.1.1 Asignación de clientes:**

Se genera la población inicial mediante heurísticas, para ello, con la ayuda del modelo de asignación se asignan los clientes, se actualizan y se iteran con al AGCB para seleccionar los mejores y con esto garantizar que cada uno de ellos tiene la información individual.

### 9.1.2 Evaluación y generación de la nueva población:

Se eligen las mejores soluciones después de aplicar algoritmos de mejoramiento, de cruce y mutación, para así escoger las más relevantes y poder llegar al punto en el cual se de satisfacción a la obtención de una solución adecuada, esto se muestra detalladamente en el ejemplo solucionado en la sección anterior.

### 9.1.3 Algoritmo genético de Chu-Beasley:

Se consideran los siguientes parámetros para resolver el problema de ruteo de vehículos aplicando el algoritmo genético de Chu-Beasley.

- a) Número de generaciones
- b) Número de individuos
- c) Porcentaje de mutación
- d) Número de individuos seleccionados por torneo
- e) Puntos de recombinación

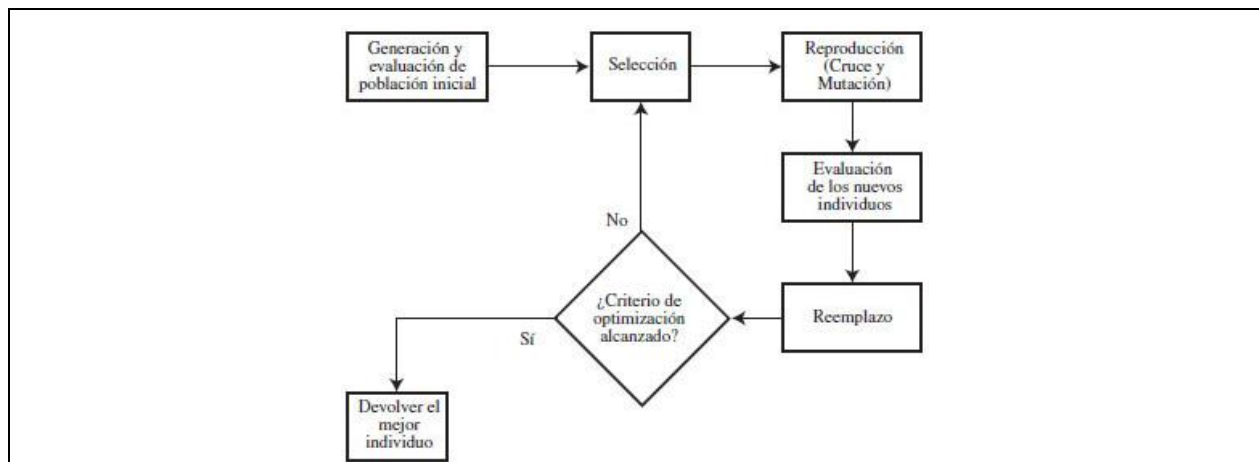


fig. 16. Diagrama de flujo del AGCB

Tomado de: <https://goo.gl/sN8P55>. Portal web dedicado a la investigación científica.



Los parámetros se calibran mediante la aplicación de algoritmo simple y el de Chu - Beasley, teniendo en cuenta que la diferencia entre ellos radica en la generación de la población que en el AGS dicha población es homogénea y en el AGCB no lo es, puesto que siempre somete a torneo a padres diferentes obteniendo así hijos distintos para ingresarlos a la población, para esto se hace necesario correrlo varias veces, e ir obteniendo los diferentes resultados para con ellos poder ir graduando cada una de las medidas.

En este sentido los parámetros se calibran ejecutando varias corridas del algoritmo, por ello la presente investigación, fundada en el problema de aplicación del ruteo de vehículos en el MDVRPTW a las zonas turísticas del departamento de Nariño, haciendo las corridas de forma independiente a cada depósito, tomará como referencia el proceso llevado a cabo en el ruteo de vehículos dependientes del tiempo [1], en donde se calibran dichos parámetros a través de la variación de estos en cada depósito.

Cabe resaltar que es posible tomar de forma independiente cada depósito, puesto que el algoritmo de asignación garantiza el intercambio de clientes entre ellos. [17][1].

A continuación, en la tabla 20 se indica el resumen de algunas corridas del algoritmo genético con valores en la función objetivo para diferentes parámetros. Se toma para todos los casos como población a 20 individuos, de los cuales, algunos son generados por heurísticas y otros seleccionados de forma aleatoria.

**Tabla 20**  
**Pruebas para población y generaciones**

<b>Corridas</b>	<b>Población</b>	<b>Generaciones</b>	<b>Tiempo computacional promedio (seg)</b>	<b>F.O Promedio</b>	<b>Desviación</b>	<b>Error promedio %</b>
	<b>Aleatoria</b>					
1	10	500	2.163	603.3573	-5.35	1.7
2	20	500	3.065	625.4802	-6.34	5.95

3	30	500	1.368	613.2041	-4.27	4.73
4	100	500	3.333	622.3548	-6.32	4.37
5	200	500	3.749	636.7146	-6.21	6.46
6	10	1000	1.548	598.9714	-7.21	3.69
7	20	1000	4.013	614.4351	-7.25	4.56
8	30	1000	1.548	620.0864	-2.35	3.87
9	100	1000	3.163	598.4632	-2.29	1.97
10	200	1000	4.988	597.1263	-3.47	3.22
11	10	1500	1.983	596.8056	-2.09	3.34
12	20	1500	3.990	594.1288	-2.18	2.90
13	30	1500	2.306	591.2326	-2.54	0.75
14	100	1500	4.013	4502.282	-2.65	4.05
15	200	1500	4.197	610.1405	-5.34	3.71
16	10	1800	1.127	598.9714	-3.65	3.52
17	20	1800	4.988	596.9020	-2.54	3.19
18	30	1800	1.070	580.6372	-2.14	0.67
19	100	1800	1.368	595.7084	-2.74	2.84
20	200	1800	3.536	596.3800	-7.32	3.11
21	10	2000	3.749	597.0677	-3.54	2.89
22	20	2000	3.333	598.0824	-2.1	2.57
23	30	2000	1.319	589.2389	-2.61	1.7
24	100	2000	2.173	607.8513	-2.6	4.93
25	200	2000	4.865	595.6018	-2.08	2.98

**Tabla 21****Número de individuos seleccionados para el torneo**

INDIVIDUOS	GENERACIONES	% ERROR		
		K=1	K=2	K=3

30	1800	3.15	1.02	4.12
30	2000	4.83	3.74	2.94

Según los resultados anteriores, los parámetros con los cuales se debe trabajar el algoritmo genético son: 30 individuos y realizando 1800 iteraciones, según [22], se puede tomar un valor del 5% para realizar la mutación.

En cuanto al número de individuos seleccionados para el algoritmo de torneo se seleccionan aleatoriamente a dos padres de la población ( $K=2$ ), y se obtiene el valor de su función objetivo, se realiza el mismo procedimiento en la segunda iteración, para con estos 4 padres seleccionados, realizar combinaciones entre ellos, para así poder generar hijos y tomar a los dos mejores, a los cuales se les realiza el mejoramiento y se presentan a la población para decidir si pueden o no ingresar.

## 9.2 Instancias

Utilizamos instancias para probar el algoritmo desarrollado en nuestro caso utilizaremos las diseñadas por Cordeau [21], considerando hasta 250 clientes, distribuidos en 3 y 4 depósitos. Se obtiene el valor la función objetivo, el error relativo, la asignación de los clientes a cada depósito, las rutas en cada uno de ellos y los gráficos correspondientes realizando para ello 10 corridas para el caso de prueba.

La nomenclatura de las instancias encontradas en [23] [21], se muestra en la tabla 13 y se describe a continuación:

En la primera fila de datos contiene la siguiente información:

Tipo: 6 correspondiente al MDVRPTW

m: número de vehículos

n: número de clientes

t: número de depósitos

Las siguientes filas contienen información para cada depósito en este caso:

D: máxima duración de la ruta

Q: máxima carga del vehículo

Las siguientes filas para cada cliente contienen la siguiente información:

i: Número de clientes

x: coordenada en x

y: coordenada en y

d: duración del servicio

q: demanda de cada cliente

f: frecuencia de la visita

a: número de combinaciones de cada visita

list: Lista de todas las combinaciones posibles de visitas

e: Comienzo de las ventanas de tiempo

l: Fin de la ventana de tiempo

**Tabla 22**

**Datos Instancia pr02 Cordeau**

Tipo 6	m = 3	n = 96	d =									
MDVRPTW			4									
D = 480	Q =195											
D = 480	Q =195											
D = 480	Q =195											
D = 480	Q =195											
i=1	x =	y =	d =	q =	f =	a =	List	e =	l =	8	410	539
	33.558	30.750	12	4	1	4	= 1	2	4			
2	48.828	65.314	1	12	1	4	1	2	4	8	125	278
...												
49	68.884	-40.54	21	22	1	4	1	2	4	8	439	540
50	76.312	86.760	5	21	1	4	1	2	4	8	216	357
...												
95	27.600	13.934	19	21	1	4	1	2	4	8	85	192
96	52.832	50.684	21	10	1	4	1	2	4	8	124	269
97	6.229	10.590	0	0	0	0	0	1000				

98	32.663	44.730	0	0	0	0	0	1000
99	48.807	48.792	0	0	0	0	0	1000
100	33.179	-4.96	0	0	0	0	0	1000

### 9.3 Resultados obtenidos para las instancias Cordeau

A continuación, en la tabla 14 se presenta los resultados obtenidos con el algoritmo desarrollado para la instancia Pr02 desarrollada por Cordeau para el problema MDVRPTW

**Tabla 23**  
**Mejores resultados para la instancia Pr02**

Caso de Prueba: Pr02

Mejor solución encontrada:

m: Número de vehículos: 4

n: Número de clientes: 50

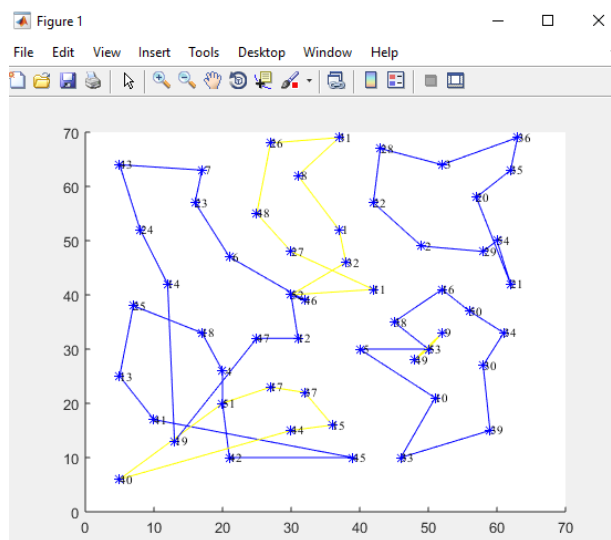
d: Número de depósitos: 4

q: Capacidad máxima del vehículo 80

ITERACION	FUNCION OBJETIVO	ERROR RELATIVO
1	480,6	0,39690829
2	482,9	0,87737623
3	484,2	1,14894506
4	487,8	1,90098183
5	482,4	0,77292668
6	487,4	1,81742219
7	483,3	0,96093587
8	478,8	0,02088991
9	487,4	1,81742219
10	491,0	2,56945895

PROMEDIO:	484,58	1,22832672
DESV. ESTÁNDAR	3,74308251	0,78192657

Los resultados resumidos en la tabla 23, muestra que el algoritmo desarrollado llega a la mejor solución en esta instancia diseñada por Cordeau en la iteración 8, y se aproxima con un porcentaje de error permisible del 2%, es decir, el algoritmo probado con esta instancia que tiene 50 clientes con 4 depósitos si es viable y apropiado para solucionar el caso de prueba de los sitios turísticos, el cual considera 3 depósitos y 33 clientes.



**fig. 17. Representación de la solución Pr02 de Cordeau generada por el algoritmo de solución.**

**Fuente:** Imagen generada con el software Matlab:

La figura 17, representa una de las mejores soluciones para el problema MDVRPTW, correspondiente a la instancia Pr02 de Cordeau, obteniendo un error relativo de 2% con respecto al óptimo, el caso de prueba incluye 50 vehículos, 4 clientes y 4 depósitos con 4 vehículos máximo por cada uno de estos, sin superar una capacidad de 80 por vehículo.

A continuación, en la tabla 24, se presenta los resultados que arroja Matlab al correr el algoritmo para la instancia Pr02.

**Tabla 24**  
**Número de clientes por depósito**

<b>DEPÓSITOS</b>			
<b>D1</b>	<b>D2</b>	<b>D3</b>	<b>D4</b>
13	17	11	9

De esta manera, se obtiene el número de clientes que se asignan a cada depósito tal como se muestra en la tabla 25.

**Tabla 25**  
**Número de clientes por cada ruta**

	<b>RUTAS POR CADA DEPÓSITO</b>			
	<b>R1</b>	<b>R2</b>	<b>R3</b>	<b>R4</b>
<b>D1</b>	3	4	3	3
<b>D2</b>	6	5	6	
<b>D3</b>	5	3	3	
<b>D4</b>	5	4		

En la tabla 26 se presenta el número de clientes que atiende cada ruta, por fila según el depósito.

**Tabla 26**  
**Demanda cubierta en cada depósito**

	<b>DEMANDA EN LAS RUTAS POR CADA DEPÓSITO</b>			
	<b>R1</b>	<b>R2</b>	<b>R3</b>	<b>R4</b>
<b>D1</b>	151	45		
<b>D2</b>	160	111		
<b>D3</b>	148	29		
<b>D4</b>	133			

En la tabla 27 se presenta el total de la demanda de cada ruta, por cada fila según el depósito.

**Tabla 27**

**Secuencia de visita de los clientes en cada ruta y depósito**

<b>VECTOR DE SECUENCIA DE VISITA DE LOS CLIENTES EN CADA RUTA Y DEPÓSITO</b>																	
46	45	43	42	41	38	26	20	19	18	16	14	5					
49	48	47	44	33	32	28	27	25	24	15	13	12	9	8	7	2	
51	50	40	39	35	34	31	17	11	10	6							
37	36	30	29	23	22	21	4	3									

En la tabla 23 se presenta la ruta que debe seguir el vehículo para visitar los clientes, teniendo en cuenta la tabla 21, por ejemplo, el cliente 46, 45, 43 corresponden a la ruta 1 del depósito 1, los clientes 42, 41, 38, 26 corresponde a la ruta 2 del primer depósito y así sucesivamente hasta completar todas las rutas de todos los depósitos.

#### **9.4 Caso de aplicación**

Después de validar el algoritmo mediante las pruebas realizadas con las instancias diseñadas por Cordeau, como se muestra en los resultados anteriores, se realiza para el caso de las rutas turísticas del departamento de Nariño diferentes corridas hasta lograr un porcentaje mínimo de error relativo, empleando para ello la misma nomenclatura de las tablas anteriormente expuestas (Tablas 15 - 18), teniendo en cuenta que nuestro caso de aplicación tiene 3 depósitos y 33 clientes.

A continuación, en la tabla 28 se presentan las matrices de distancia correspondientes a cada depósito, y teniendo en cuenta la zona.

**Tabla 28**

**Matriz de costo para el depósito 1**

	1	2	3	4	5	6	7	8	9	10	11
1	0	10,8	37,7	24,6	48,3	43,1	48,9	27,6	27,7	68,6	81,8
2	10,8	0	48,9	35	57,8	52	59,3	36,5	36,6	17,3	90,7



<b>3</b>	37,7	48,9	0	58,8	84,2	78,9	83,1	58,2	63,5	109	118
<b>4</b>	24,6	35	58,8	0	42,7	67,1	24,3	53,8	51,7	21,2	62,8
<b>5</b>	48,3	57,8	84,2	42,7	0	43	19,1	69,7	20,9	54,5	33,5
<b>6</b>	43,1	52	78,9	67,1	43	0	62,1	64,5	22,4	49,3	64,9
<b>7</b>	48,9	59,3	83,1	24,3	19,1	62,1	0	78,1	40	45,4	39,3
<b>8</b>	27,6	36,5	58,2	53,8	69,7	64,5	78,1	0	48,9	41,8	103
<b>9</b>	27,7	36,6	63,5	51,7	20,9	22,4	40	48,9	0	33,9	54,4
<b>10</b>	68,6	17,3	109	21,2	54,5	49,3	45,4	41,8	33,9	0	120
<b>11</b>	81,8	90,7	118	62,8	33,5	64,9	39,3	103	54,4	120	0

En la tabla 29 se presenta la, matriz de costos para el depósito 1. (Pasto), en donde, cada número corresponde a los siguientes municipios del departamento.

**Tabla 29**

**Zona cubierta para el depósito 1**

NUMERO	CIUDAD
1	PASTO
2	LAGUNA DE TELPIS
3	LA COCHA
4	YACUANQUER
5	SANDONA
6	EL TAMBO
7	BOMBONA
8	CHACHUAGUI
9	PASTO
10	GUALMATAN
11	LINARES

Con los anteriores datos, la ayuda del algoritmo y de la aplicación de Google Maps, se obtiene la siguiente ruta, tabla 30.

Tabla 30

Ruta generada para el depósito 1

1	8	3	2	10	4	7	11	5	9	6	1
27,6	58,2	48,9	17,3	21,2	24,3	39,3	33,5	20,9	22,4	43,1	

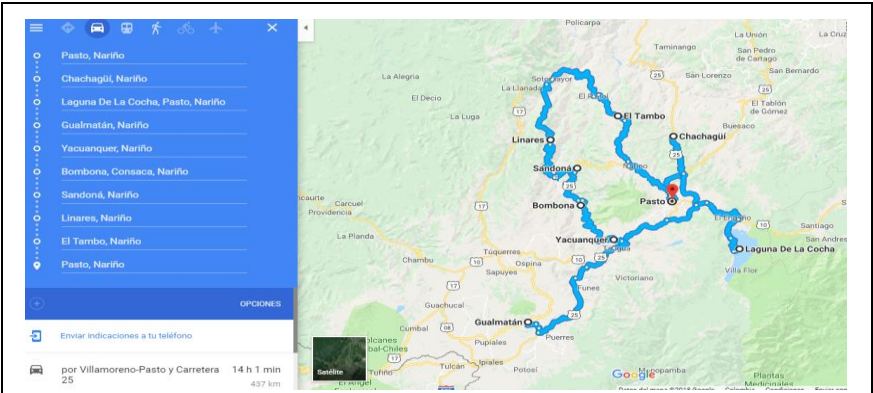


fig. 18. Ruta generada para el depósito 1. (San Juan de Pasto)

Fuente: imagen generada con Google maps.

Para el depósito 2. (Chachagüí), la zona correspondiente es la zona norte del departamento, la cual se presenta en la siguiente tabla 31.

Tabla 31

Matriz de costos para el depósito 2. (Chachagüí)

	1	2	3	4	5	6	7	8	9	10
1	0	0	55,9	62,2	38	68,8	88	148	145	59,4
2	0	0	55,9	49,8	34,1	68,8	88	159	145	58,7
3	55,9	55,9	0	80,2	53	23,2	106	114	116	56
4	62,2	49,8	80,2	0	30,5	83	84	92,6	90	50
5	38	34,1	53	30,5	0	72,5	53	123	111	39,4
6	68,8	68,8	23,2	83	72,5	0	152	90,6	93,2	232
7	87,5	87,5	106	83,9	53,4	152	0	60,8	58,2	137

8	148	159	114	92,6	123	90,6	61	0	2,6	83,7
9	145	145	116	90	111	93,2	58	2,6	0	86,3
10	59,4	58,7	56	50	39,4	232	137	83,7	86,3	0

Los datos anteriores corresponden a los siguientes puntos turísticos, tabla 32.

**Tabla 32**

**Zona cubierta para el depósito 2**

NÚMERO	CIUDAD
1	LAS MESAS LAGUNA DEL SILENCIO
2	LAS MESAS COMPLEJO VOLCÁNICO DOÑA JUANA
3	LA UNION
4	SAN LORENZO
5	BUESACO
6	SAN PABLO
7	CHACHUAGUI
8	CUMBITARA
9	TAMINANGO
10	LA UNION CERRO LA JACOBA

Con ayuda de la tabla 32 y la aplicación de Google Maps, se genera la siguiente ruta, presentada en la tabla 33:

**Tabla 33**

**Ruta generada para el depósito 2**

7	5	2	1	6	3	8	9	4	10	7
53,4	34,1	0	68,8	23,2	114	2,6	90	50	137	



Finalmente se considera la zona correspondiente al depósito 3, su matriz de costos se presenta a continuación en la tabla 34.

**Tabla 34**  
**Matriz de costo para el depósito 3. (Ipiiales)**

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	57,6	55	103	61,6	177	243	177	53,2	50,3	73	76	107
2	57,6	0	41	65,4	47,9	155	221	155	82,4	70,3	61,6	70,3	85
3	55	41	0	108	74,4	182	248	182	86,9	84	88,2	110	134
4	103	65,4	108	0	33,2	74,5	140	74,5	109	96,5	20,1	96,5	111
5	61,6	47,9	74,4	33,2	0	153	219	153	56,7	44,6	59,2	44,6	55,5
6	177	155	182	74,5	153	0	154	0	183	171	95	171	186
7	243	221	248	140	219	154	0	154	249	237	160	237	251
8	177	155	182	74,5	153	0	154	0	183	171	94,5	153	185

9	53,2	82,4	86,9	109	56,7	183	249	183	0	12,5	89	38,2	88,8
10	50,3	70,3	84	96,5	44,6	171	237	171	12,5	0	76,8	26,2	76,7
11	73	61,6	88,2	20,1	59,2	95	160	94,5	89	76,8	0	77,1	91,6
12	76	70,3	110	96,5	44,6	171	237	153	38,2	26,2	77,1	0	42,2
13	107	85	134	111	55,5	186	251	185	88,8	76,7	91,6	42,2	0

La tabla 35 corresponde a la información de cada ciudad.

**Tabla 35**  
**Zona cubierta para el depósito 3**

NÚMERO	CIUDAD
1	IPIALES – LAS LAJAS
2	VOLCAN CUMBAL
3	VOLCAN CHILES
4	RICAURTE LA PLANADA
5	TUQUERRES
6	RICAURTE
7	TUMACO
8	BARBACOAS
9	FUNES
10	EL PEDREGAL
11	MALLAMA
12	GUAITARILLA
13	SAMANIEGO

Con los datos de la tabla 35 junto con la aplicación de Google Maps, se obtiene la ruta para este depósito presentada en la tabla 36:

Tabla 36

Ruta generada para el depósito 3

1	9	12	13	5	4	6	7	8	2	3	10	10	1
53,2	38,2	42,2	55,5	33,2	74,5	154	154	155	41	84	0	50,3	



En el caso de aplicación se agrega un nuevo vector con los valores de las ventanas de tiempo, el cual es considerado como el tiempo necesario para visitar y conocer cada sitio turístico. Una vez obtenidos los resultados y según el orden de visita de los sitios turísticos, de acuerdo con la zona correspondiente a cada depósito, se presentan dichos valores resumidos en la siguiente tabla 28.

En ella se consigna el tiempo necesario para recorrer cada ruta, además de la distancia y el tiempo aproximado necesario para conocer todos y cada uno de los atractivos turísticos pertenecientes a cada una de las rutas, recomendando que se destine por lo menos un día de estancia en cada uno de ellos (en algunos casos más) ya que en cada sitio hay más lugares internos por visitar y conocer.

Por ejemplo, para el depósito Pasto, se puede hacer el recorrido de la ruta en 14 h 1 m, aproximadamente, pero con este tiempo, no se podría conocer cada uno de ellos, por eso como es lógico se hace necesaria la permanencia en cada uno de ellos para conocerlos y por eso se considera

que aproximadamente se necesitaría 480 horas, este valor corresponde a la suma de los valores del vector de ventanas de tiempo el cual permite cumplir con el objetivo de recorrer y conocer los sitios de esta ruta.

Cabe resaltar que la zona centro y sur del departamento necesitan menos tiempo para su recorrido puesto que a pesar de haber municipios lejanos el tiempo de recorrido es menor al compáralo con la zona costera, por lo que para esta ruta se necesita más tiempo de permanencia.

**Tabla 37**

**Resumen de las rutas generadas para cada depósito**

<b>DEPÓ SITO</b>	<b>RUTA</b>	<b>TIEMPO</b>		<b>DISTANCIA  Km</b>	<b>VENTANA DE TIEMPO  Horas</b>
		<b>HORA S</b>	<b>MINUT OS</b>		
Pasto	1 PASTO				
	8 CHACHAGUI				
	3 LA COCHA				
	2 GUALMATAN				
	10YACUANQUER				
	4 BOMBONA	14	1	437	480
	7 SANDONA				
	11 LINARES				
	5 EL TAMBO				
	1 PASTO				
	7 CHACHAGUI				
	5 BUESACO				
	21 LAS MESAS				
	6 SAN PABLO				

Chacha	3 LA UNION				
güí	TAJUMBINA	11	57	359	480
	8 CUMBITARA				
	9 TAMINAGO				
	4 SAN LORENZO				
	10 LA UNION				
	7 CHACHAGUI				
	1 IPIALES				
	9 FUNES				
	12 GUAITARILLA				
	13 SAMANIEGO				
Ipiiales	5 TUQUERRES	16	3	714	552
	6 RICAURTE				
	7 TUMACO				
	8 BARBACOAS				
	2 CUMBAL				
	1 IPIALES				



## X. CONCLUSIONES

La adición de la restricción que respecta a las ventanas de tiempo ha sido considerada ampliamente en otras variantes del VRP como el CVRP, el MDVRP VRPTW, el PRP, entre otros, pero escasamente en los recorridos de cada vehículo en el problema MDVRPTW concerniente al proyecto realizado.

Los algoritmos genéticos, en particular el algoritmo genético modificado de Chu-Beasley es una herramienta metaheurística, que aporta muy buenos resultados para la solución de problemas combinatoriales y en nuestro caso particular permite encontrar muy buenas soluciones al problema de ruteo de vehículos con múltiples depósitos y la restricción adicional de las ventanas de tiempo, aplicado a las rutas turísticas del departamento de Nariño, lo cual permite crecer y dar a conocer de una forma más optima el departamento puesto que se hace necesario que Nariño dependa también del turismo por sus hermosos sitios y paisajes y no lo haga únicamente de la agricultura.

El MDVRPTW es una variante del VRP y por tanto es considerado NP – HARD, esto implica que computacionalmente es difícil de resolver, por esta razón, el AGCB permite que el desgaste computacional generado a partir de correr el algoritmo durante varias generaciones, se puede evitar con el proceso de mutación el cual consiste en introducir un hijo a la población, con características diferentes a la tendencia de la solución y poder continuar por medio de combinación con la búsqueda de posibles mejores soluciones.

La efectividad de los resultados del algoritmo genético también radica en el uso del proceso de selección por torneo, puesto que esta técnica permite tomar al hijo más competitivo del algoritmo, es decir aquel hijo generado por los mejores padres y luego introducirlo a la población, garantizando que entran siempre los mejores y diferentes hijos.

El uso del aplicativo de Google Maps, permitió tener un acercamiento a la realidad de las rutas turísticas en el departamento de Nariño y sus respectivos costos. Esta aplicación permite tomar los valores de coordenadas en latitud y longitud de cada sector de interés y con base en dichas

coordenadas se encuentra los valores de distancias entre sectores, estableciendo así la matriz de costos real.

Para el caso de prueba se encontraron tres rutas principales que permiten recorrer la mayor cantidad de sitios turísticos empleando el menor tiempo y dinero posible las cuales están divididas en zonas, la sur, la zona centro y la zona norte. Cada una de éstas tienen su origen y llegada en las tres ciudades consideradas como depósitos y con ellas se puede hacer un recorrido general por el departamento y hacer un turismo diferente, dependiendo de lo que el cliente requiera.

## **XI. RECOMENDACIONES**

En la presente investigación se soluciona un problema real, y por tanto se hace necesario hacer un estudio más detallado del turismo en Nariño, teniendo en cuenta la temporada del año, puesto que esto implica que más número de visitantes lleguen al departamento y considerar también el estado de las vías puesto que esto puede llevar a la necesidad de utilizar rutas alternas para poder conocer los puntos turísticos evitando de pronto congestión en las mismas, debido a que en varias de ellas el acceso vehicular es limitado.

Se ha tenido acercamientos con empresas de turismo en Nariño, las cuales están interesadas en conocer esta propuesta, tal es el caso de “Sueño de los Andes”, a quienes les ha llamado la atención este trabajo con la idea de utilizarlo y así mejorar sus planes turísticos mediante la aplicación de la presente investigación.

Es conveniente indagar más sobre la restricción de ventanas de tiempo, puesto que son valores que pueden cambiar dependiendo del requerimiento del turista, puede ser que este tiempo mínimo no sea suficiente para la estancia en un determinado sitio.

Se recomienda realizar esta misma investigación para las rutas turísticas del departamento que se recorren en bicicleta, ya que es una necesidad en el departamento de fomentar el turismo desde otros campos y medios de transporte, por esta razón creo que se hace necesario intervenir dentro del programa de la gobernación denominado “En Bici por Nariño”, el cual permite recorrer gran parte de la región en bicicleta y por tanto, sería muy recomendable y necesario la optimización de rutas turísticas.

## XII. REFERENCIAS

- [1] F. A. Guasmayan Guasmayan, “Solución del problema de ruteo de vehiculos dependientes del tiempo utilizando un algoritmo genetico modificado”, Ttesis Maestría, Risaralda, UTP, Pereira, p. 187, 2014.
- [2] J.-F. Cordeau, G. Laporte, M. W. P. Savelsbergh, and D. Vigo, “Vehicle Routing”, Handbook in OR & MS, vol. 14.
- [3] A. Arturo, “Morada Al sur.”, *Revista Universidad Nacional de Colombia*, vol. 1, pp.1, 1945
- [4] A. C. Silva Calpa, C. R. Trejos Moncayo, D. G. Martínez Delgado, G. A. López Montezuma, "Nariño: un territotio por conocer", *Revista de Investigaciones UNAD*, vol 12, No. 2, 2013
- [5] G. A. Toro Velásquez, Fondo de Promoción Turística de Colombia, Unión temporal Turismo Huitonava, Tourism Consulting, 2012.
- [6] T. A. Camacho Caicedo, “Carnaval de Negros y Blancos,” [nariño.gov.co/Publicaciones](http://nariño.gov.co/Publicaciones), 2009, p 1-6.
- [7] A. J. Pirabán, “Problema de Enrutamiento de Vehículos,” no. Nov, pp. 2–4, Seminario de Investigación I, Maestría en Ingeniería de Sistemas y computación, Universidad Nacional de Colombia, 2008.
- [8] A. C. Silva Calpa, y C. R. Trejos Moncayo, “Prospectiva del café en Nariño,” *Revista Estratégica Organizacional/ Unad/ ISSN 2339-3866/ vol. 5/ No. 1-2/ pp. 11-28*, 2016.
- [9] Nilson Herazo Padilla, “Modelación matemática del problema de ruteo de vehículos con restricciones de múltiples depósitos, flota heterogénea de vehículos y ventanas de tiempos 1”, [repositorio.cuc.edu.co](http://repositorio.cuc.edu.co), pp. 1–147, 2012.
- [10] P. Vansteenwegen, W. Souffriau, and G. Vanden Berghe, “Metaheuristics for Tourist Trip Planning Metaheuristics for Tourist Trip Planning,” *Lecture Notes in Economics and Mathematical Systems*, no.624 May, 2016.
- [11] G. Dantzig, R. Fulkerson, and S. Johnson, “SOLUTION OF A LARGE-SCALE TRAVELING-SALESMAN PROBLEM.” *Operations Research Society of América*, p. 393 - 410, 1994
- [12] J.-F. Cordeau, G. Laporte, M. W. P. Savelsbergh, and D. Vigo, “Chapter 6 Vehicle Routing,” IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd, vol. 14, no.

- 06, pp. 367–428, 2007.
- [13] M. M. Solomon, J. Cordeau, H. Etudes, and J. Desrosiers, “The VRP with Time Windows,” *Veh. Routing Probl.*, pp. 157–193, 1999.
  - [14] H. Bae and I. Moon, “Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles,” vol. 40, pp. 6536–6549, 2016.
  - [15] F. S. Hillier and G. J. Lieberman. *Investigación de Operaciones*. Décima Edición. México DF: Stanford University, Mcgraw-Hill/Interamericana editores, S.A, 2015
  - [16] V. Probabilístico and P. M. G, “Aproximación para la Mejora Local 2-p-opt del Problema,” Artículo de investigación, Ingeniería de transporte, vol. 13, no. 1994, pp. 5–12, 1995.
  - [17] M. J. Bravo Montenegro and M. en I. O. y Estadística, “Metodología híbrida para resolver el problema de ruteo de vehículos con depósitos múltiples considerando el consumo de combustible,” Tesis Maestría, Risaralda, UTP, Pereira, 2016.
  - [18] W. A. Díaz, P. Universidad, P. Universidad, M. S. Felices, P. Universidad, and W. S. Sotillo, “Evaluación cuantitativa de rutas de distribución utilizando técnicas combinadas de optimización,” no. October, pp. 0–9, 2016.
  - [19] A. Olivera, “Heurísticas para problemas de ruteo de vehículos,” *Inst. Comput. - Fac. Ing.*, p. 63, 2004.
  - [20] E. a Bermeo Muñoz and J. H. Calderón Sotero, “Diseño de un modelo de optimización de rutas de transporte,” *El Hombre y la Máquina*, vol. 32, no. 32, pp. 52–67, 2009.
  - [21] Y. A. Hernández Ortiz. "Diseño de un sistema de Ruteo de Vehículos con múltiples depósitos en empresas de transporte de carga por carretera", M.S tesis, Cundinamarca, U. D. Francisco José de Caldas, p. 85, Bogotá, 2016
  - [22] E. M. Toro Ocampo, Gallego Rendón, y A. Escobar Zuluaga, “Técnicas Metaheurísticas de Optimización,” Universidad Tecnológica de Pereira, vol. Segunda Ed, 2008.
  - [23] S. N. Kumar and R. Panneerselvam, “A Time-Dependent Vehicle Routing Problem with Time Windows for E-Commerce Supplier Site Pickups Using Genetic Algorithm,” *Intell. Inf. Manag.*, 2015.

### XIII. ANEXOS

#### *Anexo 1*

A continuación, se presenta el programa y la instancia con la cual se lo valida y se calibran los parámetros del modelo.

```
% type m n t
% type:
% 0 (VRP)
% 1 (PVRP)
% 2 (MDVRP)
% 3 (SDVRP)
% 4 (VRPTW)
% 5 (PVRPTW)
% 6 (MDVRPTW)
% 7 (SDVRPTW)
% m: number of vehicles
% n: number of customers
% t: number of days (PVRP), depots (MDVRP) or vehicle types (SDVRP)%
% The next t lines contain, for each day (or depot or vehicle type), the following
information:%
% D Q %
% D: maximum duration of a route
% Q: maximum load of a vehicle
% The next lines contain, for each customer, the following information: %
% i x y d q f a list e l %
% i: customer number
% x: x coordinate
% y: y coordinate
% d: service duration
% q: demand
```

```

% f: frequency of visit
% a: number of possible visit combinations
% list: list of all possible visit combinations
% e: beginning of time window (earliest time for start of ervice), if any
% l: end of time window (latest time for start of service), if any
% Each visit combination is coded with the decimal equivalent of the corresponding binary
bit string. For example, in a 5-day period, the code 10 which is equivalent to the bit string
01010 means that a customer is visited on days 2 and 4. (Days are numbered from left to
right.)%
% Note : In the case of the MDVRP, the lines go from 1 to n + t and the last t entries
correspond to the t depots. In the case of the VRP, PVRP and MDVRP, the lines go from
0 to n and the first entry corresponds to the unique depot.%
% 2 4 50 4
clc
clear all
close all
% veh=4;% num de vehiculos
% clientes=50;
% depositos=4;
% % 0 80
% % 0 80
% % 0 80
% % 0 80
% cap=80; % capacidad de cada vehiculo
% M=[ 1 37 52 0 7 1 4 1 2 4 8
% 2 49 49 0 30 1 4 1 2 4 8
% 3 52 64 0 16 1 4 1 2 4 8
% 4 20 26 0 9 1 4 1 2 4 8
% 5 40 30 0 21 1 4 1 2 4 8
% 6 21 47 0 15 1 4 1 2 4 8

```

% 7 17 63 0 19 1 4 1 2 4 8  
% 8 31 62 0 23 1 4 1 2 4 8  
% 9 52 33 0 11 1 4 1 2 4 8  
% 10 51 21 0 5 1 4 1 2 4 8  
% 11 42 41 0 19 1 4 1 2 4 8  
% 12 31 32 0 29 1 4 1 2 4 8  
% 13 5 25 0 23 1 4 1 2 4 8  
% 14 12 42 0 21 1 4 1 2 4 8  
% 15 36 16 0 10 1 4 1 2 4 8  
% 16 52 41 0 15 1 4 1 2 4 8  
% 17 27 23 0 3 1 4 1 2 4 8  
% 18 17 33 0 41 1 4 1 2 4 8  
% 19 13 13 0 9 1 4 1 2 4 8  
% 20 57 58 0 28 1 4 1 2 4 8  
% 21 62 42 0 8 1 4 1 2 4 8  
% 22 42 57 0 8 1 4 1 2 4 8  
% 23 16 57 0 16 1 4 1 2 4 8  
% 24 8 52 0 10 1 4 1 2 4 8  
% 25 7 38 0 28 1 4 1 2 4 8  
% 26 27 68 0 7 1 4 1 2 4 8  
% 27 30 48 0 15 1 4 1 2 4 8  
% 28 43 67 0 14 1 4 1 2 4 8  
% 29 58 48 0 6 1 4 1 2 4 8  
% 30 58 27 0 19 1 4 1 2 4 8  
% 31 37 69 0 11 1 4 1 2 4 8  
% 32 38 46 0 12 1 4 1 2 4 8  
% 33 46 10 0 23 1 4 1 2 4 8  
% 34 61 33 0 26 1 4 1 2 4 8  
% 35 62 63 0 17 1 4 1 2 4 8  
% 36 63 69 0 6 1 4 1 2 4 8



```

% 37 32 22 0 9 1 4 1 2 4 8
% 38 45 35 0 15 1 4 1 2 4 8
% 39 59 15 0 14 1 4 1 2 4 8
% 40 5 6 0 7 1 4 1 2 4 8
% 41 10 17 0 27 1 4 1 2 4 8
% 42 21 10 0 13 1 4 1 2 4 8
% 43 5 64 0 11 1 4 1 2 4 8
% 44 30 15 0 16 1 4 1 2 4 8
% 45 39 10 0 10 1 4 1 2 4 8
% 46 32 39 0 5 1 4 1 2 4 8
% 47 25 32 0 25 1 4 1 2 4 8
% 48 25 55 0 17 1 4 1 2 4 8
% 49 48 28 0 18 1 4 1 2 4 8
% 50 56 37 0 10 1 4 1 2 4 8
% 51 20 20 0 0 0 0 0 0 0
% 52 30 40 0 0 0 0 0 0 0
% 53 50 30 0 0 0 0 0 0 0
% 54 60 50 0 0 0 0 0 0 0];
% depo=[51 20 20 0 0 0 0
% 52 30 40 0 0 0 0
% 53 50 30 0 0 0 0
% 54 60 50 0 0 0 0];
%[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=turismo
%[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=colacteosmd
[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp02 % datos
iniciales
%[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp01
%[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp04
%[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp06
%[M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp08

```

```

% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp10
% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp12
% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp14
% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp16
% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp18
% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp20
% [M,depo,veh, clientes, depositos, cap, soluciones,capdeposito]=rotartsp22
%matriz de dist
%%% llamar funcion para asignar clientes y adicionar soluciones a las
%%% rotaciones
pesovacio=1415;pesolleno=pesovacio+cap;
[sasignacion]=asignaclientes(M,veh,depositos,cap,pesolleno,pesovacio,M(:,5))
soluciones=[soluciones;sasignacion];
nciud=length(M(:,1))-depositos;

for k=1:(nciud+depositos)

    d(k,k)=0;
    for k1=k+1:(nciud+depositos) % matriz de distancias
        d(k,k1)=sqrt((M(k,2)-M(k1,2))^2+(M(k,3)-M(k1,3))^2);
        d(k1,k)=d(k,k1);
        maximadis(k)=max(d(k,:));
        if min(d(k,:))==0
            [c,d1]=min(d(k,:));
            d(k,d1)=inf;
            minimadis(k)=min(d(k,:));
            d(k,d1)=0;
        end
    end

end
end

```

end

```
%d=xlsread('matrizdistanciascmddepomov.XLSX','Hoja1','A1:EW153');
```

```
%d=xlsread('matrizdistanciasturismo.XLSX','Hoja3','A1:AG33');
```

```
% d=[0 121 135 115 48.9 92.9 35 114 92.9 112 221 57.8 89.3 58.8 287 131
52 59.3 221 96.7 63 36.6 50.7 94.1 127 76.6 118 17.3 128 90.7 10.8 101
36.5
```

```
% 121 0 41 65.4 144 196 92.7 47.9 196 215 155 135 176 162 221 234 153 117
155 200 82.4 137 70.3 197 61.6 70.3 11.9 62.9 85 89.4 110 57.6 140
```

```
% 135 41 0 108 158 209 106 74.4 209 228 182 149 190 175 248 248 167 135 182
213 86.9 151 84 211 88.2 110 29.1 64.8 134 116 124 55 153
```

```
% 115 65.4 108 0 170 222 119 33.2 238 241 74.5 161 202 188 140 260 179 145
74.5 226 109 164 96.5 223 20.1 96.5 230 112 111 115 136 103 166
```

```
% 48.9 144 158 170 0 114 58.8 137 114 133 245 84.2 95 80.2 311 153 78.9 83.1
457 118 86.7 63.5 74.3 116 151 100 142 109 151 118 37.7 124 58.2
```

```
% 92.9 196 209 222 114 0 111 190 0 55.9 297 126 62.2 38 363 68.8 121 135 297
148 139 106 126 145 203 152 59.4 98.6 203 160 84 176 87.5
```

```
% 35 92.7 106 119 58.8 111 0 85.5 110 129 193 42.7 90.5 75.8 259 148
67.1 24.3 193 114 34.9 51.7 22.4 111 97.5 48.5 90 21.2 99.1 62.8 24.6
72.7 53.8
```

```
% 114 47.9 74.4 33.2 137 190 85.5 0 170 189 153 110 166 136 219 208 127
91.2 153 174 56.7 112 44.6 171 59.2 44.6 50 79.2 55.5 113 103 61.6 114
```

```
% 92.9 196 209 238 114 0 110 170 0 55.9 297 126 49.8 34.1 363 68.8 121 135
297 159 139 106 126 145 203 152 58.7 98.6 203 160 93.7 176 87.5
```

```
% 112 215 228 241 133 55.9 129 189 55.9 0 316 145 80.2 53 382 23.2 139 154
316 114 158 125 145 116 222 171 56 117 222 179 103 194 106
```

```
% 221 155 182 74.5 245 297 193 153 297 316 0 236 277 262 154 335 253 218 0 300
183 238 171 298 95 171 153 187 186 190 211 177 240
```

```
% 57.8 135 149 161 84.2 126 42.7 110 126 145 236 0 123 92 302 164 43 19.1 236
130 77.5 20.9 65 127 142 91.1 133 54.5 142 33.5 48.3 138 69.7
```

% 89.3 176 190 202 95 62.2 90.5 166 49.8 80.2 277 123 0 30.5 359 83 118  
 132 294 92.6 135 102 123 90 200 149 50 95 200 166 80.5 156 83.9  
 % 58.8 162 175 188 80.2 38 75.8 136 34.1 53 262 92 30.5 0 329 72.5 87 101  
 263 123 105 71.6 92.3 111 169 118 39.4 64.5 169 126 49.9 142 53.4  
 % 287 221 248 140 311 363 259 219 363 382 154 302 359 329 0 400 319 283 154 366  
 249 304 237 364 160 237 219 253 251 256 277 243 306  
 % 131 234 248 260 153 68.8 148 208 68.8 23.2 335 164 83 72.5 400 0 159 173  
 353 90.6 177 144 165 93.2 241 191 232 200 241 198 122 214 152  
 % 52 153 167 179 78.9 121 67.1 127 121 139 253 43 118 87 319 159 0 62.1 253  
 125 95 22.4 82.6 122 159 109 126 49.3 159 64.9 43.1 158 64.5  
 % 59.3 117 135 145 83.1 135 24.3 91.2 135 154 218 19.1 132 101 283 173 62.1  
 0 217 138 59.1 40 46.7 136 124 72.7 114 45.4 123 39.3 48.9 96.9 78.1  
 % 221 155 182 74.5 457 297 193 153 297 316 0 236 294 263 154 353 253 217 0 300  
 183 288 171 298 94.5 153 153 187 185 190 211 177 240  
 % 96.7 200 213 226 118 148 114 174 159 114 300 130 92.6 123 366 90.6 125 138  
 300 0 142 109 130 2.6 207 156 83.7 102 2017 163 87.8 180 60.8  
 % 63 82.4 86.9 109 86.7 139 34.9 56.7 139 158 183 77.5 135 105 249 177  
 95 59.1 183 142 0 79.7 12.5 139 89 38.2 79.7 37.9 88.8 97.6 52.6 53.2  
 82.1  
 % 36.6 137 151 164 63.5 106 51.7 112 106 125 238 20.9 102 71.6 304 144 22.4  
 40 288 109 79.7 0 67.2 107 144 93.2 135 33.9 144 54.4 27.7 117 48.9  
 % 50.7 70.3 84 96.5 74.3 126 22.4 44.6 126 145 171 65 123 92.3 237 165  
 82.6 46.7 171 130 12.5 67.2 0 127 76.8 26.2 67.6 36.7 76.7 85.2 40.2  
 50.3 69.7  
 % 94.1 197 211 223 116 145 111 171 145 116 298 127 90 111 364 93.2 122 136 298  
 2.6 139 107 127 0 204 153 86.3 99.6 204 161 85.2 177 58.2  
 % 127 61.6 88.2 20.1 151 203 97.5 59.2 203 222 95 142 200 169 160 241 159  
 124 94.5 207 89 144 76.8 204 0 77.1 59 92.8 91.6 96.1 106 73 146

```

% 76.6 70.3 110 96.5 100 152 48.5 44.6 152 171 171 91.1 149 118 237 191 109
72.7 153 156 38.2 93.2 26.2 153 77.1 0 67.5 60.6 42.2 43.1 66.2 76
95.7
% 118 11.9 29.1 230 142 59.4 90 50 58.7 56 153 133 50 39.4 219 232 126
114 153 83.7 79.7 135 67.6 86.3 59 67.5 0 60.4 82.4 86.9 91.8 183 137
% 17.3 62.9 64.8 112 109 98.6 21.2 79.2 98.6 117 187 54.5 95 64.5 253
200 49.3 45.4 187 102 37.9 33.9 36.7 99.6 92.8 60.6 60.4 0 111 120
68.6 24.3 41.8
% 128 85 134 111 151 203 99.1 55.5 203 222 186 142 200 169 251 241 159 123 185
2017 88.8 144 76.7 204 91.6 42.2 82.4 111 0 28 117 107 146
% 90.7 89.4 116 115 118 160 62.8 113 160 179 190 33.5 166 126 256 198 64.9
39.3 190 163 97.6 54.4 85.2 161 96.1 43.1 86.9 120 28 0 81.8 172 103
% 10.8 110 124 136 37.7 84 24.6 103 93.7 103 211 48.3 80.5 49.9 277 122
43.1 48.9 211 87.8 52.6 27.7 40.2 85.2 106 66.2 91.8 68.6 117 81.8 0
90.2 27.6
% 101 57.6 55 103 124 176 72.7 61.6 176 194 177 138 156 142 243 214 158 96.9
177 180 53.2 117 50.3 177 73 76 183 24.3 107 172 90.2 0 120
% 36.5 140 153 166 58.2 87.5 53.8 114 87.5 106 240 69.7 83.9 53.4 306 152
64.5 78.1 240 60.8 82.1 48.9 69.7 58.2 146 95.7 137 41.8 146 103 27.6
120 0];

```

```
d=d/100;
```

```
contcamiones=1;
```

```
for ns=1:3
```

```
maxdeM=max(maximadis);
```

```
mindeM=min(minimadis);
```

```
solincu=inf;
```

```
n=1;
```

```
P=zeros(1,nciud);
```

```
f=[0];
```

```

vfob=601;
%while vfob>600

    while n<=3000 % num de iteraciones
        %seleccion de circulo o elipse rotada
        % selec=[1:1:8]; 1: circulo, 2 :elipse sin rotar....
        angulo=zeros(1,depositos);
        % numero aleatorio entre los valores maximo y minimo
        for i=1:depositos
            escalador=0.001; % escala para rango de elipses
            ale(1,i)=round(9*rand(1)+1);
            ale(2,i)=round(9*rand(1)+1);
            angulo(i)=round(180*rand(1));
            radioaleatorio(1,i)=round((mindeM+(ale(1,i)-0)*(maxdeM-mindeM)/(10-
0))/(escalador));
            radioaleatorio(2,i)=round((mindeM+(ale(2,i)-0)*(maxdeM-mindeM)/(10-
0))/(escalador));
        end

        %circulos
        s=zeros(depositos,nciud);% variable que contriene los puntos correspondientes a los
        cirulos
        acum=0;
        for ii=1:nciud%examinar cada punto
            percir=0;
            %radioaleatorio=[];
            per=zeros(1,depositos);
            dis=[];
            disi=[];
            for k=1:depositos

```

```

dis(k)=inf;
disi(k)=inf;%distancia en caso de interseccion
end
flagintersecto=zeros(1,depositos);
%capacidad de los depositos
%acum=acum+M(ii,5);
for j=1:depositos
    %%%%%%%%%% solucionar sistema de ecuacioens
%    format short
    [sx sy]=solecu2(M,depo,ii,j,radioaleatorio(1,j),radioaleatorio(2,j),angulo(j));
    % calcular distancia desde el dep al punto intersectado
    dist=abs(sqrt(sx^2+sy^2));
    %%%%%%%%%%
    %if sqrt((depo(j,2)-M(ii,2))^2+(depo(j,3)-M(ii,3))^2)<=radioaleatorio(j)
    if sqrt((M(ii,2)-depo(j,2))^2+(M(ii,3)-depo(j,3))^2)<=dist %pertenece al circulo
        flagintersecto(j)=1;
        %per(j)=i+1;
        disi(j)=sqrt((depo(j,2)-M(ii,2))^2+(depo(j,3)-M(ii,3))^2);
        %percir=1;
        %elseif percir~=1 %bandera
    else
        % eliminar rotacion de la interseccion sx y sy
        dis(j)= sqrt((depo(j,2)-M(ii,2))^2+(depo(j,3)-M(ii,3))^2)-dist;%dist del pto al
borde de la elipse
    end
    if (j==depositos)
        if(sum(flagintersecto)==0)
            [a,b]=min(dis);
            per(b)=ii+1;%matriz de 1 fila
        else

```

```

        [a,b]=min(disi);
        per(b)=ii+1;
    end
end
end
for m=1:depositos
    if per(m)==ii+1
        s(m,ii)=ii+1;%contiene los puntos de cada dep en m filas igual a dep
    end
end
end
end

% pertenencia de los puntos a los circulos
ss=[];% matriz con filas igual al numero de circulos por columnas igual al
% numero de puntos para almacenar los puntos de mayor a menor
% correspondientes
Camiones=[];%variable aux para eliminar los ceros de ss
Pobla=zeros(depositos,nciud);
for i=1:depositos
    orden=sort(s(i,:));%ordenar de menor a mayor
    ceros=numel(orden(orden==0));% numero de ceros en el vector
    Camiones(i,:)=[length(s(i,:))-ceros,0];
    ss(i,:)=rot90(orden)';%ordena de mayor a menor
    % calcular TSP de los puntos seleccionados
    % se tiene en cuenta que los depositos estan en las ultimas filas
    Ruta=ss(i,1:Camiones(i,1));
    D=[d(nciud+i,:);d];% matriz de distancias con el deposito correspondiente en la primera
    fila
    col=D(:,(nciud+i)); %Columna correspondiente al deposito
    D=[col,D];%Columna del deposito en la primera posicion

```



```

D=D(1:(nciud+1),1:(1+nciud));% se eliminan las filas y columnas de ...
%los otros depositos dejando solo el deposito correspondiente
[pobla fo(i)]=calcularTSP1(Ruta,Camiones(i,:),D,pesolleno,pesovacio,M(:,5));
Pobla(i,1:length(pobla))=pobla;
end

%   foincu=sum(fo);
%   if foincu<solincu
%   solincu=foincu;
%   poblaincu=Pobla;
%   end

P=[P;Pobla];
f=[f;sum(fo)];
    n=n+1;
end

P=P(2:length(P),:);% rutas
f=f(2:length(f));
[vm cm]=min(f);% valor mminimo y su posicion de la fun obj.del TSP
% Mp=P(cm*4-3:cm*4,:);
% mmp=Mp(1,:);
% cerosruta=numel(mmp(mmp==0));% numero de ceros en el vector
% ruta=mmp(1,1:(length(mmp)-cerosruta));
% camiones=[length(ruta),0];

Maxsize=100;
MaxDis=100000;
figure(1);
hold on
n=nciud;

```

```

% ciclo para envio a funcion del VRP geneticos
solucionfinal=zeros(length(soluciones(:,1))+1,nciud);
vecdepo=[];% guardar el numero de clientes de cada deposito
veccamiones=zeros(1+length(soluciones(:,1))*depositos,nciud);
Mp=P(cm*depositos-(depositos-1):cm*depositos,:);% son las rutas en orden por cada
deposito
marca=1;
for h=1:depositos

mmp=Mp(h,:);
cerosruta=numel(mmp(mmp==0));% numero de ceros en el vector
ruta=mmp(1,1:(length(mmp)-cerosruta));% clientes correspondientes a cada depo
camiones=[length(ruta),0];% el numero de ciudades de cada deposito
coordenadas=[depo(h,2);depo(h,3)];% coordenadas del deposito
coordenadas=[coordenadas,M(1:nciud,2:3)'];
demanda=[0;M(:,5)];
% capacidad de los vehiculs, los depositos presentan capacidad homogenea
% llana a funcion de condiciones iniciales para VRP
% [RUTA                                     camiones
fobje(h)]=vrpgeneticos(ruta,camiones(1),coordenadas,demanda(1:nciud+1),1.001*cap,M
,depositos)% variables que se envian:
[RUTA                                     camiones
fobje(h)]=vrpgeneticos(ruta,camiones(1),coordenadas,demanda(1:nciud+1),1.01*cap,M,
depositos)% variables que se envian:
vfob(1)=sum(fobje);
solucionfinal(1,marca:length(RUTA)+marca-1)=RUTA;
marca=marca+length(RUTA);
vecdepo(1,h)=length(RUTA);
veccamiones(h,1:length(camiones))=camiones;
% ruta: clientes de cada depo

```

[illegible]

```

camionesfin=veccamiones;
%%% hacer mejoramiento shift para cada sol
%mejoramiento con shift 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

conc=1;
cfin=[];
fo=[];
cfin=[];
dfin=[];
sfin=[];
for sf=1:length(solucionfinal(:,1))
    solfin=solucionfinal(sf,:);
    solfin=solfin-1;
    for i=1:depositos
        ncer(i)=numel(camionesfin(camionesfin(i,:)==0));
    end
    camfin=camionesfin(:,1:nciud-min(ncer)+5);

    depofin=vecdepo(sf,:);
    [fo(sf) cfin(conc:depositos*sf,:) dfin(sf,:) sfin(sf,:)] =
    mshift(M,depo,depofin,camfin(conc:depositos*sf,:),min(vfob),solfin,cap,pesolleno);
    conc=1+depositos*sf;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% obtener la mejor solucion para dibujarla
[fobjmin,posiminfo]=min(fo);
%posiminfo=4;

```

```

fobjmin
solfin=sfin(posiminfo,:);
depofin=dfin(posiminfo,:)
camionesfin=cfin(posiminfo*depositos-depositos+1:posiminfo*depositos,:)%esta
completado con ceros hasta nciud
%%%% funcion para graficar
figure(1)
marcar=1;
solfin=solfin+1;
for b=1:depositos
    Dis=[depo(b,2),M(1:nciud,2)';depo(b,3),M(1:nciud,3)'];
    camionesceros=camionesfin(b,:);
    soldibujar=solfin(marcar:marcar+depofin(b)-1);
    marcar=marcar+depofin(b);
    dibujarVRP(soldibujar,Dis,depofin(b),camionesceros);
end
plot(M(:,2),M(:,3),'*b');
solfin=solfin-1;

demrutas=zeros(length(camionesfin),length(camionesfin));
cont=0;a=1;
for i=1:nciud
    demandaruta(i)=M(solfin(i),5);
end
for i=1:depositos
    j=1;
    while camionesfin(i,j)~=0
        aux(j)=camionesfin(i,j);
        demrutas(i,j)=sum(demandaruta(a:a-1+aux(j)));%demanda de cada ruta
        a=aux(j)+a;
    end
end

```

```

j=j+1;
end
end
text(M(:,2),M(:,3),num2str((1:length(M(:,1)))),'%0.0f'),'fontname','times','fontsize',8)
toc

```

*Anexo 2.* Iteraciones con Matlab; Al correr el programa, algunos de las rutas son:

**Tabla 38**  
**Rutas generadas con el código en Matlab**

RUTAS									
RUTA									
1	2	4	5	6	7	10	9	8	3
RUTA									
2	30	3	23	29	4	37	36	21	22
RUTA									
3	3	4	5	6	7	10	9	8	2

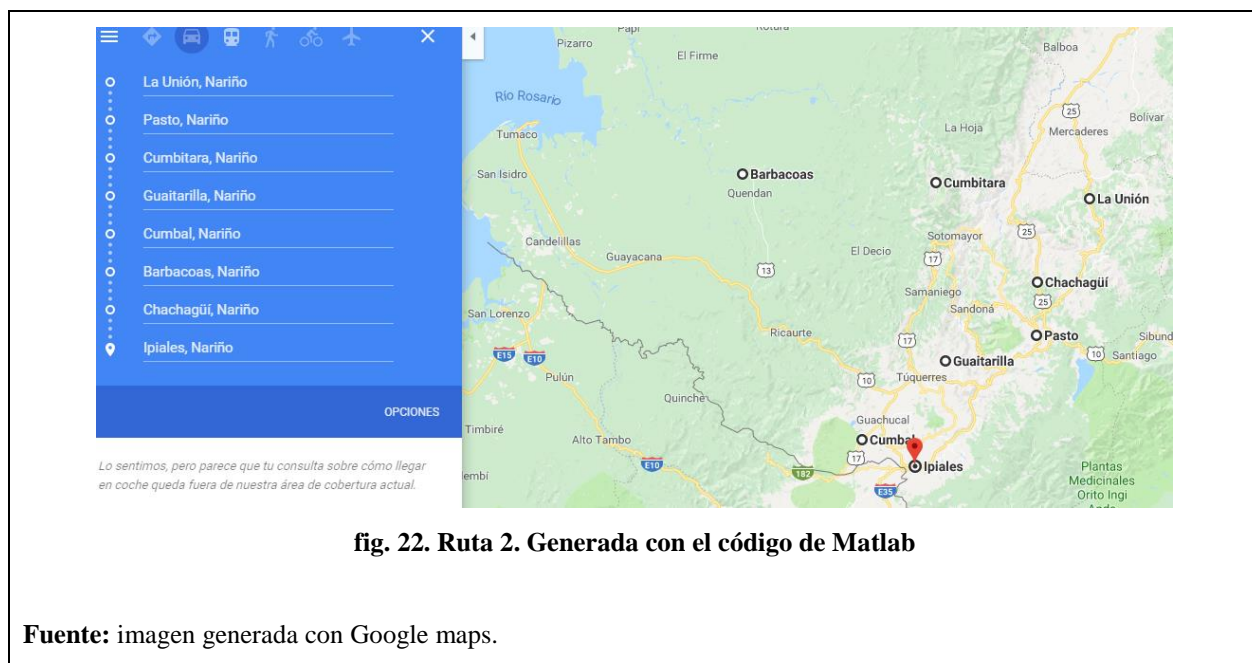




fig. 23. Ruta 3. Generada con el Código de Matlab.

Fuente: imagen generada con Google maps.

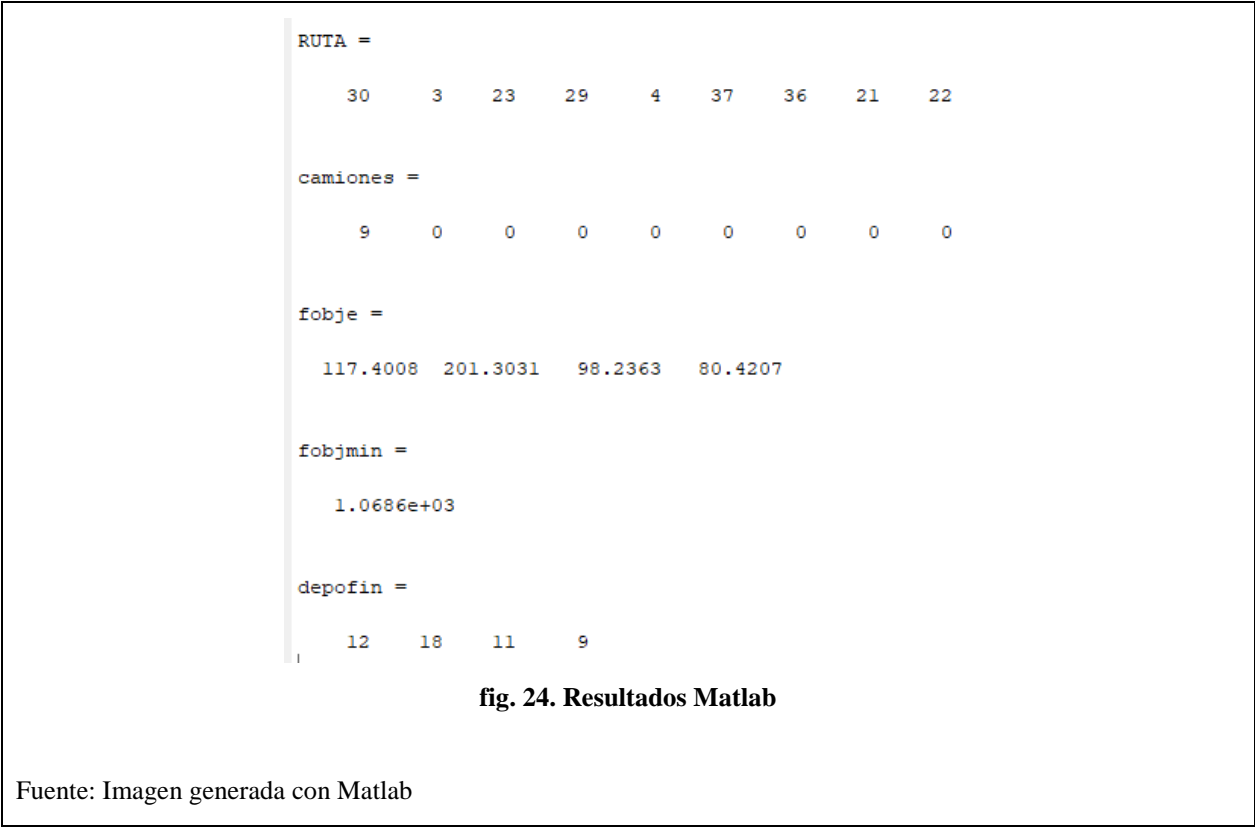


fig. 24. Resultados Matlab

Fuente: Imagen generada con Matlab



**Anexo 3 y 4****Tabla 39****Datos de la demanda de cada sitio turístico**

CIUDAD	HOTEL	COMIDA	PASAJES/		DIAS*			
	(\$)	(\$)	INTERNOS	SUMA	#PERSONAS	#DÍAS	PERSONAS	DEMANDA
PASTO	92000	26000	30000	148000	1	3	3	444000
IPIALES	79000	28000	22000	129000	1	2	2	258000
PASTO	92000	26000	10000	128000	1	2	2	256000
CUMBAL	25000	25000	32000	82000	1	2	2	164000
CUMBAL	25000	25000	48000	98000	1	3	3	294000
RICAUARTE	15000	20000	50000	85000	1	3	3	255000
LA COCHA	80000	38000	20000	138000	1	2	2	276000
LAS MESAS	20000	20000	200000	240000	1	2	2	480000
YACUANQUER	20000	20000	0	40000	1	1	1	40000
TUQUERRES	30000	20000	16000	66000	1	2	2	132000
LAS MESAS	20000	20000	200000	240000	1	2	2	480000
UNION	80000	25000	120000	225000	1	2	2	450000
RICAUARTE	15000	20000	100000	135000	1	2	2	270000
SANDONA	20000	20000	0	40000	1	1	1	40000
SAN LORENZO	20000	18000	50000	88000	1	1	1	88000
BUESACO	50000	20000	8000	78000	1	2	2	156000
TUMACO	170000	60000	100000	330000	1	7	7	2310000
SAN PABLO	15000	22000	4000	41000	1	1	1	41000
EL TAMBO	15000	18000	0	33000	1	1	1	33000
BOMBONA	20000	20000	6000	46000	1	1	1	46000
BARBACOAS	30000	26000	80000	136000	1	1	1	136000
CHACHUAGUI	70000	28000	18000	116000	1	3	3	348000
CUMBITARA	20000	20000	80000	120000	1	2	2	240000
FUNES	15000	18000	40000	73000	1	1	1	73000
PASTO	92000	26000	16000	134000	1	2	2	268000
EL PEDREGAL	80000	25000	4000	109000	1	2	2	218000

TAMINANGO	50000	30000	20000	100000	1	2	2	200000
MALLAMA	25000	20000	45000	90000	1	1	1	90000
GUAITARILLA	20000	20000	20000	60000	1	1	1	60000
LA UNION	80000	25000	40000	145000	1	3	3	435000
GUALMATAN	25000	20000	20000	65000	1	2	2	130000
SAMANIEGO	22000	20000	0	42000	1	3	3	126000
LINARES	23000	20000	40000	83000	1	2	2	166000

**Anexo 5****Demanda por zonas y capacidad****Tabla 40****Demanda por zonas y capacidad**

	SUMA DEMANDA i	CAPACIDAD PERSONA	DE LA
PASTO	2047000	3070500	
IPIALES	1007000	1510500	
TUQUERRES	1090000	1635000	
LA UNION	2570000	3855000	
TUMACO	686000	1029000	

**Anexo 6.**

**TITULO** Metaheuristics for Tourist Trip Planning

**AUTOR(ES)** Pieter Vansteenwegen, Wouter Souffriau, Greet Vanden Berghe,  
and Dirk Van Oudheusden  
University of Leuven  
Chapter in Lecture Notes in Economics and Mathematical Systems

**FECHA DE** 2009

**PUBLICACIÓN**

<b>OBJETIVO</b>	Presentar una visión general de las metaheurísticas utilizadas en el turismo, seleccionando los puntos de utilidad más interesantes y diseñando un viaje turístico personalizado.
<b>RESUMEN</b> <b>(QUE, COMO,</b> <b>PARA QUE)</b>	El objetivo de este trabajo es presentar una visión general de las metaheurísticas utilizadas en el turismo e introducir una búsqueda de vecindad variable sesgada para resolver el problema de orientación de equipo (TOP). Seleccionando los puntos de interés más interesantes y diseñando un viaje turístico personalizado, se puede modelar como un TOP con ventanas de tiempo (TOPTW). La búsqueda local guiada (GLS) y la búsqueda variable de la vecindad (VNS) se aplican para resolver eficientemente el TOP. La búsqueda local iterada (ILS) se implementa para resolver el TOPTW. Los algoritmos GLS y VNS se comparan con las heurísticas más conocidas y se aplican en grandes conjuntos de problemas. Los resultados obtenidos son casi de la misma calidad que los resultados de estas heurísticas, pero el tiempo computacional se reduce significativamente. Para algunos de los problemas VNS calcula nuevas mejores soluciones. Los resultados del algoritmo de ILS, aplicado a grandes conjuntos de problemas, tienen un intervalo promedio con la solución óptima de sólo el 2,7%, con mucho menos esfuerzo computacional.
<b>ANÁLISIS</b> <b>CRÍTICO</b>	<b>MODELO PRESENTADO EN EL ARTÍCULO:</b> La importancia del anterior artículo radica en el objeto de estudio: el turismo, para el cual se utilizan varias heurísticas y metaheurísticas, además se muestra cómo se las aplica para trazar viajes turísticos personalizados a los puntos con mayor interés. <b>MODELO PROPUESTO EN ESTA INVESTIGACIÓN:</b> La investigación está encaminada al diseño de un mapeo utilizando entre otras la aplicación Google maps, para el cual se necesitaría encontrar las mejores rutas desde tres diferentes depósitos a cada sitio turístico del departamento, las cuales se determinarán como

viajes turísticos personalizados, utilizando para ello la solución del problema de ruteo de vehículos con múltiples depósitos y ventanas de tiempo (MDVRPTW), con la ayuda de la búsqueda local iterada (ILS), La idea principal de búsqueda local iterativa es tratar de generar vecinos que nos generen pozos de atracción vecinos (o al menos diferentes)

**CRITERIO DE BUSQUEDA** <https://www.researchgate.net/publication/226088125>

**TITULO** Modelo de ruteo para generar rutas turísticas

**AUTOR(ES)** BEATRIZ ILIANA YAÑEZ MANCILLA

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

**FECHA DE** 2013

**PUBLICACIÓN**

**OBJETIVO** Desarrollar un modelo de optimización, mediante la adaptación del problema de ruteo de vehículos, para generar rutas turísticas que maximicen la satisfacción del cliente, sujeto a restricciones de presupuesto y tiempo.

**RESUMEN (QUE, COMO, PARA QUE)** En este texto se propone un modelo de ruteo para la administración adecuada de los recursos del cliente turístico y satisfacerlo a partir de una variedad de rutas personalizadas en un tiempo breve de espera, en consecuencia, el objetivo de esta investigación es desarrollar un modelo de optimización, mediante la adaptación del problema de

<b>ANÁLISIS CRÍTICO</b>	<p>ruteo de vehículos, para generar rutas turísticas que maximicen la satisfacción del cliente, con restricciones de presupuesto y tiempo.</p> <p><b>MODELO PRESENTADO EN EL ARTÍCULO:</b> Este artículo muestra la adaptación del sistema de ruteo para lograr un modelo de optimización y con ello generar rutas turísticas.</p> <p><b>MODELO PROPUESTO EN ESTA INVESTIGACIÓN:</b> A comparación de dicha investigación, se optimizarán las rutas turísticas del departamento de Nariño, adaptando el problema de ruteo de vehículos con ventanas de tiempo y generando así un mapeo con las mejores rutas.</p>
<b>CRITERIO BUSQUEDA</b>	<p><b>DE</b> <a href="http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/6563/Modelo%20de%20ruteo%20para%20generar%20rutas%20tur%C3%Adsticas.pdf?sequence=1">http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/6563/Modelo%20de%20ruteo%20para%20generar%20rutas%20tur%C3%Adsticas.pdf?sequence=1</a></p>
<b>TITULO</b>	<p>Heuristics for the time dependent team orienteering 117 problem: Application to tourist route planning</p>
<b>AUTOR(ES)</b>	<p>Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, Grammati Pantziou, Nikolaos Vathis</p> <p>GRECIA</p>
<b>FECHA PUBLICACIÓN</b>	<p><b>DE</b> 2015</p>
<b>OBJETIVO</b>	<p>El objetivo principal de este problema es seleccionar POIs que se ajusten a las preferencias turísticas, teniendo en cuenta la gran variedad de parámetros y restricciones</p>
<b>RESUMEN (QUE, COMO, PARA QUE)</b>	<p>El problema de orientación del equipo dependiente del tiempo con ventanas de tiempo (TDTOPTW) puede usarse para modelar varios problemas de la vida real. Entre ellos, el problema de planeación de rutas para los turistas interesados en visitar varios puntos de interés (POI) utilizando el transporte público. El</p>

objetivo principal de este problema es seleccionar POIs que coincidan con las preferencias turísticas, teniendo en cuenta la gran variedad de parámetros y restricciones, respetando el tiempo disponible para la vista diaria y dinamizando el transporte público para viajar entre POIs (TTDP) Mientras que casi todo el cuerpo de la literatura relacionada se ocupa de la versión dependiente del tiempo del problema. La única heurística TDTOPTW propuesta para la base de datos no se basa en los horarios de los servicios de transporte periódico. Proponemos una eficiente heurística basada en clústeres para el TDTOPTW que proporciona soluciones de alta calidad, toma en cuenta la dependencia del tiempo en el cálculo de los tiempos de viaje entre POIs y asume la hipótesis en los horarios periódicos de servicio. Y conjuntos de datos POI reales compilados del área metropolitana de Atenas (Grecia). Nuestros algoritmos TTD mano de aprendizaje (es decir, tiempo de atención determinada) en lugar de fijo inicio / final lugares para los itinerarios turísticos derivados.

## **ANÁLISIS CRÍTICO**

## **MODELO MODELO PROPUESTO EN PRESENTADO EN EL ESTA INVESTIGACIÓN:**

### **ARTÍCULO:**

En el artículo se muestra la aplicación de una poderosa y eficiente heurística basada en clústeres aplicada al problema de orientación de equipo dependiente del tiempo con ventanas de tiempo (TDTOPTW) la cual proporciona soluciones de calidad, tomando en cuenta

Se aplicará el problema de ruteo de vehículos con ventanas de tiempo y asociarlo con los sitios turísticos del departamento de Nariño, para encontrar los mejores recorridos, no los puntos turísticos de mayor interés, sino las rutas óptimas para visitar cada uno empelando el menor tiempo y dinero posible.

la dependencia del tiempo en el cálculo de los tiempos de viaje entre puntos de interés y poder con ello seleccionar los puntos de interés que coincidan con las preferencias turísticas.

**CRITERIO DE BUSQUEDA** <http://www.sciencedirect.com/science/article/pii/S0305054815000817>

**TITULO** A heuristic for the location of a rapid transit line  
**AUTOR(ES)** Giuseppe Bruno, Michel Gendreau, Gilbert Laporte.  
 Universidad de Napoli,  
 Departamento de informática y sistemas  
 Italia

**FECHA DE PUBLICACIÓN** 2000

**OBJETIVO** Presentar un modelo matemático y una heurística de dos fases para la localización de un tránsito rápido en un entorno urbano.

**RESUMEN (QUE, COMO, PARA QUE)** Este artículo presenta un modelo matemático y una heurística de dos fases para la localización de una alineación de tránsito rápido en un entorno urbano. Este método se puede ver como un bloque de construcción para el problema de diseño de red multilínea. Los resultados computacionales sobre instancias generadas al azar y sobre algunos datos reales de Milán conllevan la eficiencia del enfoque propuesto.

**ANÁLISIS CRÍTICO** **MODELO PRESENTADO EN EL ARTÍCULO:** **MODELO PROPUESTO EN ESTA INVESTIGACIÓN:**

En esta investigación se encuentra una alineación de De acuerdo con lo anterior, se pretenderá encontrar una línea de

tránsito rápido en un entorno urbano, utilizando un modelo matemático y una heurística en dos fases. tránsito rápido para cada uno de los sitios turísticos de departamento y trazar el mapa con las mejores rutas.

**CRITERIO DE BUSQUEDA** <http://www.sciencedirect.com/science/article/pii/S0305054800000514>

**TITULO** Planning rapid transit networks

**AUTOR(ES)** G. Laporte, J.A. Mesa, F.A. Ortega, F. Perea. España

**FECHA DE** 2011

**PUBLICACIÓN**

**OBJETIVO** revisar algunos índices para la calidad de una red de tránsito rápido, así como modelos matemáticos y heurísticos que pueden ser utilizados para diseñar redes.

**RESUMEN (QUE, COMO, PARA QUE)** Los proyectos de construcción de tránsito rápido son esfuerzos importantes que requieren la planificación a largo plazo de varios actores, incluyendo políticos, planificadores urbanos, ingenieros, consultores de administración y grupos de ciudadanos. Tradicionalmente, los métodos de investigación operativa no han desempeñado un papel importante a nivel de planificación, pero varias herramientas desarrolladas en los últimos años pueden ayudar al proceso de decisión y ayudar a producir diseños de red provisionales que se pueden presentar a los planificadores para una evaluación posterior. Este artículo revisa algunos índices para la calidad de una red de tránsito rápido, así como modelos matemáticos y heurísticos que pueden ser utilizados para diseñar redes.

**ANÁLISIS CRÍTICO** **MODELO PRESENTADO EN EL ARTÍCULO:** **MODELO PROPUESTO EN ESTA INVESTIGACIÓN:**



Analizando el presente De este modo se escogerá el documento, se revisan y se mejor modelo y la mejor muestran algunas de las herramienta que permita herramientas y modelos construir el mapeo con las matemáticos utilizados para mejores rutas de tránsito rápido producir diseños de red de a cada uno de los sitios tránsito rápido turísticos del departamento de Nariño.

**CRITERIO DE BUSQUEDA** [www.sciencedirect.com/science/journal/00380121](http://www.sciencedirect.com/science/journal/00380121)

**TITULO** Optimization of the Multi-Depot & Multi-Vehicle Pickup and Delivery Problem with Time Windows using Genetic Algorithm

**AUTOR(ES)** E. Ben Alaïa, I. Harbaoui Dridi, H. Bouchriha, P.Borne. Francia

**FECHA DE PUBLICACIÓN** 2013

**OBJETIVO**

Presentamos en este trabajo una breve revisión bibliográfica del MDPDPTW y construimos un modelo matemático del problema anterior. Presentamos nuestro enfoque basado en algoritmos genéticos para dar un conjunto de soluciones factibles

**RESUMEN (QUE, COMO, PARA QUE)**

En este artículo se presenta un algoritmo genético (GA) para el problema de multi-vehículo y multi-depósitos de recogida y entrega con ventanas de tiempo (m-MDPDPTW). El m-MDPDPTW es una clase de problema de enrutamiento de vehículos que debe construir un conjunto de rutas para satisfacer las peticiones de transporte, entre los nodos de origen (clientes de recogida) y los nodos de destino (clientes de entrega), satisfaciendo las restricciones de precedencia, capacidad y tiempo. Cada carga debe ser transportada por uno de los vehículos, entre los clientes de recogida y entrega. Asumimos que todos los vehículos tienen la misma capacidad y que cada uno comienza y finaliza la ruta

en el mismo depósito. Presentamos en este trabajo una breve revisión bibliográfica del MDPDPTW y construimos un modelo matemático del problema anterior. Presentamos nuestro enfoque basado en algoritmos genéticos para dar un conjunto de soluciones factibles

**ANÁLISIS CRÍTICO**      **MODELO PRESENTADO EN EL ARTÍCULO:**      **MODELO PROPUESTO EN ESTA INVESTIGACIÓN:**

**ARTÍCULO:** Se acondicionará un modelo matemático para aplicar el problema de ruteo de vehículos con matemático basado en múltiples depósitos y ventanas de tiempo para algoritmos genéticos para cimentar un conjunto de rutas optimas que dar un conjunto de permita visitar la mayor cantidad de sitios turísticos del departamento de Nariño, problema de enrutamiento de utilizando la menor cantidad de tiempo y vehículos multi-depósito y dinero posibles.

multi-vehículo de recogida y entrega con ventanas de tiempo (MDPDPTW), que debe construir un conjunto de rutas para satisfacer las peticiones de transporte.

**CRITERIO DE BUSQUEDA**      <http://ieeexplore.ieee.org.ezproxy.unal.edu.co/stamp/stamp.jsp?tp=&arnumber=6689568>

**TITULO**      Solución del problema de ruteo de vehículos dependientes del tiempo utilizando un algoritmo genético modificado

**AUTOR(ES)**      Guasmayan Guasmayan Fredy Alexander

**FECHA DE**      2014

**PUBLICACIÓN**

**OBJETIVO** Desarrollar e implementar una metodología basada en el Algoritmo Genético modificado que permita establecer las rutas de los vehículos en el problema de ruteo de vehículos dependientes de tiempo, teniendo en cuenta como función objetivo la minimización de los costos requeridos en la ejecución de todas las visitas desde el depósito.

**RESUMEN** Con este proyecto se pretende resolver el problema TDVRP haciendo uso de varias heurísticas que permitan generar soluciones iniciales para el mejoramiento en búsqueda de una solución más eficiente por medio de una metaheurística denominada algoritmo genético modificado de Chu-Beasley

<b>ANÁLISIS CRÍTICO</b>	<p><b>MODELO PRESENTADO EN EL ARTÍCULO:</b></p> <p>Desarrollar e implementar una metodología basada en el Algoritmo Genético modificado que permita establecer las rutas de los vehículos en el problema de ruteo de vehículos dependientes de tiempo, teniendo en cuenta como función objetivo la minimización de los costos requeridos en la ejecución de todas las visitas desde el depósito.</p>	<p><b>MODELO PROPUESTO EN ESTA INVESTIGACIÓN:</b></p> <p>Desarrollar una metodología que permita asociar el problema de ruteo de vehículos con múltiples depósitos y ventanas de tiempo, para encontrar las mejores rutas que permitan visitar la mayor cantidad de sitios turísticos del departamento de Nariño empleando menos tiempo y dinero.</p>
-------------------------	--	---

**CRITERIO DE BUSQUEDA** <http://repositorio.utp.edu.co/dspace/handle/11059/4562>

**TITULO** Optimization methods for the planning of rapid transit systems

**AUTOR(ES)** 1

Gilbert Laporte, Juan A. Mesa, Francisco A. Ortega

**FECHA DE** 1998

**PUBLICACIÓN**

**OBJETIVO** Este artículo revisa los principales métodos disponibles para solucionar el problema de planificar los sistemas de tránsito rápido.

**RESUMEN** Una cuestión central al planificar los sistemas de tránsito rápido es la  
**(QUE, COMO,** determinación de alineaciones y estaciones. Los métodos de  
**PARA QUE)** investigación operativa pueden ayudar a resolver estos problemas y también son útiles para la evaluación de las características de la red. Este artículo de la encuesta revisa los principales métodos disponibles.

**ANÁLISIS** **MODELO PRESENTADO EN** **MODELO PROPUESTO EN**  
**CRÍTICO** **EL ARTÍCULO:** **ESTA INVESTIGACIÓN.**

Se revisan los principales métodos disponibles que permitan dar solución a la planificación de sistemas de tránsito rápido. Encontrar el mejor método de tránsito rápido que permita generar un mapeo del departamento de Nariño y con él se pueda visitar el mayor número de sitios turístico en el menor tiempo posible, comprobar cuál de los métodos propuestos se ajusta mejor para realizar dicho mapeo.

**CRITERIO DE** <http://www.sciencedirect.com/science/article/pii/S0377221799000168>  
**BUSQUEDA**